

Universitat de Lleida
Escola Politècnica Superior
Enginyeria Tècnica en Informàtica de Sistemes

Treball de final de carrera

Disseny i gamificació d'un portal esportiu

Autor: Jordi Virgili Gomà

Directora: Rosa Gil

Agost 2013

Agraïments:

No se ben bé com hauria de començar, ja que aquest apartat donaria per una tesis sencera. Així que citant als grans llibres, “Al principi”:

Al meu pare que m’ha inculcat la curiositat per saber, sense tenir por de quant de temps em portarà ... perquè “són cinc minuts”. A la meva mare, per confiar en mi sempre, sense entendre gaire bé de que va el tema però esforçant-se per entendre-ho.

A la Rosa, que m’ha guiat en aquest nou i misteriós camí que ha estat el projecte; al Robert, que ha fet de co-tutor en alguna ocasió tot i no ser-ho formalment; i al seu petit ajudant, que escoltava sense protestar les meves dubtes. Gràcies per compartir amb mi els teus pares una estona, tens una gran família.

Al grup de *Trolls*, pels bons anys en la nostra UdL (dins i fora de les aules) i per obrir-me les mires a altres punts de vista, en especial al Jhonny (Si, se que John no s’escriu així) que em va posar en contacte amb els tutors.

I per últim però no menys important, a la Maite, que em recorda que no tot són línies de codi en la vida (i menys mal!)

Índex de continguts

Índex de continguts	3
Índex de figures	7
Bloc 1:	9
Introducció.....	9
1.1 <i>Gimnàs Bordeta</i> , Qui son?:.....	10
1.2 Motius de l'elecció d'aquest tema:	11
1.3 Estructura del document:.....	13
1.4 Objectius:	14
1.5 Riscos:.....	17
1.6 Planificació:	19
1.7 Costos:.....	25
Bloc 2:	27
Estat	27
2.1 Metodologia:	27
2.1.1 Requeriments:	28
2.1.1.1 Tipus de centre:.....	28
2.1.1.2 Usuaris:.....	29
2.1.1.3 Necessitats especials:.....	32
2.2 Tecnologies:.....	36
2.2.1 Llenguatges:	36
2.2.2 Programari utilitzat pel desenvolupament:	37
2.2.3 Característiques de la plataforma d'execució:.....	39
Bloc 3:	40
Disseny	40
3.1 Arquitectura de la informació:	40
3.1.1 Card Sorting:.....	40
3.1.2 Perfils d'usuaris:	42
3.1.3 Casos d'ús i especificacions:.....	43

3.1.3.1 Casos d'ús Portal:	44
3.1.3.2 Casos d'ús Administrador:.....	47
3.1.3.3 Casos d'ús Gamificació:	57
3.2 Prototipatge:	64
3.2.1 Prototip de paper:	65
3.2.2 Proves de color:.....	67
3.2.3 Prototip Digital:	69
3.2.3.1 Colors:	70
3.2.3.2 Logotips:	72
3.2.3.3 Tipografia:	73
3.2.3.4 Organització:	73
3.3 Previ a implementar:.....	74
Bloc 4:	76
Desenvolupament	76
4.1 Disseny del portal:.....	76
4.1.1 Tema:.....	77
4.1.1.1 Esquema de colors:	77
4.1.1.2 Tipologia:	79
4.1.1.3 Logotips i botons:	80
4.1.1.4 Ordenació dels blocs:	87
4.1.1.5 Disseny Responsiu:.....	90
4.1.2 Mòduls d'autoria externa:.....	97
4.1.2.1 Editor WYSIWYG:.....	98
4.1.2.2 Simple Gmaps:.....	99
4.1.2.3 Slideshows:.....	99
4.1.2.4 Calendari:	101
4.1.2.5 Contacta:	103
4.1.2.6 Capcha:.....	104
4.1.2.7 Encriptat de dades:	105
4.2 Disseny de mòduls:	106
4.2.1 Admin gym:	106
4.2.1.1 Info:	106
4.2.1.2 Install:	107

4.2.1.3 Module:	118
4.2.1.3.1 Nous clients	119
4.2.1.3.2 Assistència	122
4.2.1.3.3 Gestionar usuari de competició	124
4.2.1.3.4 Gestionar pagament.....	127
4.2.1.3.5 Informació usuari	129
4.2.2 Admin gym funcions extra:	133
4.2.2.1 Info:	133
4.2.2.2 Module:	134
4.2.2.2.1 Actualitzar cinturó	135
4.2.2.2.2 Actualitzar dades federació.....	137
4.2.2.2.3 Consultar dades bancàries	140
4.2.3 Gamifica:	143
4.2.3.1 Info:	144
4.2.3.2 Install:	144
4.2.3.3 Module:	152
4.2.3.3.1 Afegir entrada: campionat abdominals.....	154
4.2.3.3.2 Afegir entrada: campionat peses	156
4.2.3.3.3 Afegir entrada: campionat bicicleta estàtica	158
4.2.3.3.4 Afegir entrada: campionat squash	160
4.2.3.3.5 Resultats campionat abdominals	165
4.2.3.3.6 Resultats campionat peses.....	167
4.2.3.3.7 Resultats campionat bicicleta estàtica	169
4.2.3.3.8 Resultats campionat squash.....	171
4.2.4 Mòdul Usuari:.....	173
4.2.4.1 Info:	173
4.2.4.2 Install:	174
4.2.4.3 Module:	176
4.2.4.3.1 Nova Fita	177
4.2.4.3.2 Associar fita	179
4.2.4.3.3 Gaming	182
4.3 Disseny de pàgines estàtiques que compleixen funcionalitats:.....	189
Bloc 5:	191

Conclusions	191
5.1 Usabilitat:	191
5.1.1 Navegadors:.....	191
5.1.2 TAW:.....	195
5.1.2 Usuaris:.....	196
Bloc 6:	197
Bibliografia	197
Annex A: Enquestes realitzades	198

Índex de figures

Figura 1: Logotip Gimnàs Bordeta	10
Figura 2: Antiga web	11
Figura 3: Nova Web	12
Figura 4: Fases de desenvolupament	27
Figura 5: Esquema de colors del portal	33
Figura 6: Proves de visió	34
Figura 7: Casos d'ús portal	44
Figura 8: Casos d'ús Administrador	47
Figura 9: Casos d'ús Gamificació	57
Figura 10: Prototip en Paper	65
Figura 11: Prototip en paper: Horaris	66
Figura 12: Prototip en Paper: Compta d'usuari	66
Figura 13: Proves de color (Fred)	67
Figura 14: Proves de color (Càlid)	68
Figura 15: Proves de color (Banner gris)	68
Figura 16: Proves de color (Final)	69
Figura 17: Interfície d'Axure	70
Figura 18: Export d'Axure en HTML	70
Figura 19: Banners	71
Figura 20: Logotip	72
Figura 21: Medalles	73
Figura 22: Bartik	77
Figura 23: Esquema de colors	77
Figura 24: Enlaceboton	80
Figura 25: Boto	81
Figura 26: Botobanner	82
Figura 27: Menús desplegable	87
Figura 28: Blocs a Inhabilitar	88
Figura 29: Login d'usuari	88
Figura 30: Asistent Addthis.com	89
Figura 31: Compartir Addthis.com	90
Figura 32: Logos seguir publicacions	90
Figura 33: No responsiu	91
Figura 34: Responsiu	97
Figura 35: Editor de textos	98
Figura 36: Simple GMaps	99
Figura 37: Slideshow	101
Figura 38: Horaris	102
Figura 39: Events	103
Figura 40: Formulari de contacte	104
Figura 41: Re-Captcha	105

Figura 42: <i>Navegadors Web (Konqueror)</i>	192
Figura 43: <i>Navegadors Web (Epiphany)</i>	193
Figura 44: <i>Internet Explorer 10</i>	194
Figura 45: <i>Internet Explorer 9</i>	194
Figura 46: <i>Internet Explorer 7</i>	195
Figura 47: <i>Anàlisi WCAG2.0</i>	196

Bloc 1:

Introducció

Aquest treball té com a objectiu proporcionar una base informàtica sòlida per a un centre esportiu, des d'un simple portal web bàsic per promocionar-se i donar informació sobre els serveis que ofereixen, passant per la gestió administrativa dels usuaris (i la seva informació de federacions, gestió de pagaments al centre...) i incloent la opció de gamificar els serveis que ofereixen com a novetat en aquestos tipus de serveis.

En aquest apartat s'introduirà l'empresa familiar en la que s'ha realitzat l'aplicació pràctica (*Gimnàs Bordeta*), es descriurà els motius que han portat a la realització d'aquest treball, i per acabar s'especificarà l'estructura del document presentat.

1.1 Gimnàs Bordeta, Qui son?:



Figura 1: Logotip Gimnàs Bordeta

Des de fa més de 20 anys els integrants de la família Catalina gestionen les els diferents grups que conformen *el Gimnàs Bordeta*.

Actualment, disposen d'unes instal·lacions recentment renovades i més àmplies en el barri de La Bordeta de Lleida on s'ofereixen un gran i variat espectre de possibilitats per gaudir de l'esport i d'una vida sana, amb un ampli equip de professionals especialitzats dins de les seves àrees i amb anys d'experiència docent.

El *Gimnàs Bordeta* és un espai familiar, dirigit per gent jove altament qualificada i amb una màxima clara: fer del centre un lloc on la pràctica de l'activitat física i la promoció de la salut siguin primordials, però fomentant el bon ambient i mai sense oblidar les relacions personals. Tot i haver crescut en mida i nombre d'assistents, continua sent bàsicament un negoci familiar i proper als clients.

1.2 Motius de l'elecció d'aquest tema:

L'autor d'aquest projecte ha estat l'últim any participant com a client en algunes de les activitats que ofereix el centre. En aquestes activitats, el control de l'assistència s'efectuava escrivint els noms en una fulla de paper. Com a futur titulat en *ETIS*, li va sobtar el fet de que no sols es realitzés aquesta a mà, sinó que totes les gestions del gimnàs es fessin en suport paper, sens cap tipus de base de dades (únicament una fulla d' Excel per a la informació de facturació dels clients).

Per buscar informació, la pàgina web (que encara es pot consultar a www.gimnasbordeta.com, degut a problemes amb l'antic *webmaster*) era molt deficient, tant en continguts (on no donava quasi cap tipus d'informació, alguns enllaços no funcionaven, etc.) com en aspecte (realitzada en HTML pla, estètica general d'inicis dels 90', amb barres de desplaçament fins i tot en les imatges). A més feia molt de temps que no s'actualitzava.



Figura 2: Antiga web

Amb la possibilitat de millorar aquesta situació (i amb l'autor tenint que escollir un tema pel realitzar el projecte de final de carrera), es va posar en contacte amb els administradors del gimnàs per tal de comprovar si els

interessava la construcció d'un portal nou, des d'on a més es poguessin fer totes les gestions que realitzaven en paper.

Des de *Gimnàs Bordeta* van voler tirar endavant aquesta idea, ja que a més feia temps que buscaven a algú per informatitzar l'apartat d'administració, així que es va procedir a buscar un tutor pel projecte. Al ser de Sistemes, l'alumne no disposava d'un clar candidat a qui demanar consell, així que va recórrer a companys de Gestió, que havien realitzat un projecte similar.

La tutora recomanada va resultar tot un encert, amb molta empenta i noves idees per aplicar. D'ella es la idea de gamificar el portal, de manera que aquells que vulguin se'ls doni l'opció de convertir en un joc les diferents activitats que es desenvolupen en el centre.

Al Nadal de l'any 2013, es va pujar una primera versió del portal web per substituir a la pagina web abans esmentada, únicament amb la informació bàsica, a la qual s'han anat acoblant mòduls (alguns d'ells realitzats per a l'ocasió, alguns adaptats i uns altres de ja disponibles a internet) per a realitzar totes les funcions projectades en aquest document.



Figura 3: Nova Web

1.3 Estructura del document:

El present document consta del següents blocs, amb els continguts especificats a continuació:

- **Bloc 1: Introducció.**

En aquest bloc es troba tota la informació referent al tema que s'ha escollit pel projecte, en quin espai s'aplicarà la teoria, com es va idear la proposta, l'ordre que segueix aquest document i els objectius i funcionalitats que vol assolir.

- **Bloc 2: Estat.**

Per a dur a terme aquest projecte, s'han utilitzat una sèrie de tecnologies diferents. En el segon bloc es farà referència als estudis realitzats als usuaris a qui va destinat el servei, el qual ens han permès establir les tecnologies amb les que es realitzarà el projecte.

- **Bloc 3: Disseny.**

Amb els estudis previs analitzats, s'expliquen els diferents casos d'us que es pretenen i com s'ha dissenyat la solució, conformant el primer prototip funcional, primer sobre paper i després en suport digital.

- **Bloc 4: Desenvolupament.**

Aquest es el punt on la teoria xoca amb la realitat. En aquest bloc es mostra la resposta que s'ha ideat per satisfer les necessitats i dificultats detectades en els anteriors apartats, tant en disseny de base de dades com en implementació de codi i funcionalitats. Al acabar el desenvolupament es realitzen una sèrie de proves de funcionalitat i posada a punt.

- **Bloc 5: Conclusions.**

Tant el projecte finalitzat com el procés seguit es analitzat amb ull crític i algunes millores aplicables salten a la llum.

- **Bloc 6: Bibliografia.**

Acabem el document citant la bibliografia que ens ha facilitat la feina, tant durant la investigació com en el desenvolupament.

- Annex A: Enquestes realitzades.

1.4 Objectius:

Després d'un seguit de reunions entre l'autor i el personal del gimnàs que hauria d'utilitzar-lo, es van establir els següents punts a tenir en compte:

Punt 1· El portal ha de ser la cara visible del gimnàs en la web on qualsevol usuari sense identificar pugui trobar tota la informació (actualitzada) que pugui ser d'interès del gimnàs a oferir.

Punt 2· Les actualitzacions del portal han de poder ser realitzades per personal que no es dediqui a la informàtica, per tal de canviar ràpidament els continguts i fer la pàgina més dinàmica.

Punt 3· Les dades sensibles que publiquin els membres del gimnàs en l'àrea administrativa haurà de gaudir d'una protecció extra.

Punt 4· La interfície haurà de complir amb el disseny acurat i multidispositiu (*responsive design*) i per tant funcional independentment del dispositiu, navegador i/o resolució de pantalla.

Respectant aquests punts, es va procedir a realitzar una enquesta als usuaris, qui al cap i a la fi es a qui va dirigit el 70% de l'aplicació (la resta és per a administració del centre). Les estadístiques extretes (que s'analitzaran al final del següent bloc) van ser molt reveladores, i afectaven directament a alguns d'aquests punts.

El 45% dels usuaris enquestats tenien entre 15 i 25 anys, i el 25% en tenien menys de 15 (degut a que s'imparteixen classes de *taekwondo* infantil), del total el 87% tenen un telèfon mòbil amb capacitat de navegar per internet (segurament degut a que el 75% dels enquestats tenien menys de 25 anys). Per tant, el punt 4 era molt important, ja que un gran nombre de usuaris accedirien a les funcionalitats des del telèfon mòbil, mentre estessin al centre. De la mateixa manera, els entrenadors no poden disposar d'un

ordenador amb teclat i ratolí en les diferents sales (per qüestions d'espai i ordre), així que utilitzarien *tablets* per introduir les dades.

La solució que s'ofereixi al gimnàs ha de tenir un disseny multidispositiu.

A més a més, la majoria dels enquestats que havien entrat a la pagina web del gimnàs, ho havien fet per buscar informació sobre els horaris (que no estaven actualitzats, cosa que respectant el punt 2 no hagués passat, ja que cada monitor podria actualitzar les dades de les activitats de les que es responsabilitza) i la resta per trobar informació sobre activitats que es realitzaven (no gaire ben explicades).

La solució que s'ofereixi al gimnàs ha de permetre que algú inexpert modifiqui i agregui continguts.

Com es diu en el punt 1 el portal web ha de ser el centre de publicació de totes les dades que la gent pugui necessitar.

La solució ha de passar per un procés de determinar les tasques que pot desitjar un usuari.

La part que més els interessava al personal del gimnàs era la d'administració, tot i que no tenien molt clar com portar-la a terme ni si tindrien que canviar molt la seva metodologia de treball. Així la base de dades es realitza seguint el mateix esquema que s'utilitzava al gimnàs en paper, però posant èmfasis en la seguretat (punt 3) i afegint funcionalitats noves.

La solució que s'ofereixi al gimnàs ha de ser segura i limitar la exposició de dades dels usuaris.

Un dels elements més de més novetat d'aquest treball és la gamificació de les tasques que es desenvolupen durant el dia a dia per als usuaris que ho escullin. El personal del gimnàs no te massa clara la implementació, així que ho deixen a l'elecció dels desenvolupadors d'aquest treball, tot i que pensen que ben portat pot ser un punt positiu per a ser escollits entre gimnasos de la mateixa gamma per clients nous i per a augmentar l'interès dels clients actuals en tenir més constància.

I és que un dels objectius d'aquest treball es el de demostrar que es pot gamificar qualsevol àmbit del dia a dia.

Utilitzarà el clàssic sistema *PBL (Points, Badges & Leaderboards)*, àmpliament descrit en diversos assajos sobre gamificació, com ara *For the Win (how game thinking can revolutionize your business)* escrit per Kevin Werbach i Dan Hunter.

Els punts es rebran de dues maneres, vessants típiques dels sistemes de jocs:

- Assoliment d'objectius individuals (en totes les activitats).
- Esports de competició directa (Com els partits d'*squash*).

En els primers, es proposaran un seguit d'actes o activitats concretes, l'assoliment de les quals assignarà un nivell determinat, els quals seran lineals (establint una escala de nivells que va des de principiant fins a expert), tot i que també es pretén crear un sistema de nivells no lineal, de manera que la consecució de certes fites es vegin reflexes en el rang que ostenta l'usuari.

En el segon, s'utilitzarà el típic sistema dels jocs de lluita, amb un rang basat en la *ratio* de confrontacions oficials guanyades/perdudes (definites per l'entrenador i amb un entrenador present).

A més, l'assoliment d'objectius entremetjats no necessaris per augmentar el nivell (i per tant motiu d'excel·lència en la disciplina, o almenys de notorietat), seran premiats amb fites que estaran a la pagina publica del usuari que els hagi aconseguit.

Com a últim element, en tots els esports mesurables d'alguna manera (km totals en bici, repeticions d'exercici i fins i tot assistència al centre) s'establiran multitud de taules classificatòries, per confrontar als usuaris de manera pública.

1.5 Riscos:

Durant el desenvolupament del projecte es pot topar amb contingències diverses. Si es preveuen possibles riscos, es pot minimitzar els danys que provoquen sobre la resta del projecte.

Relacionat amb les eines de desenvolupament, el riscos més grans són:

- Desconeixement parcial dels llenguatges i que s'hauran d'utilitzar durant el desenvolupament de l'aplicació.
- Desconeixement total d'algunes de les eines que s'han d'utilitzar durant la tasca, i els conseqüents dubtes de si altres eines complirien millor les expectatives.

Aquests dos riscos resulten inevitables. La ignorància dels llenguatges únicament pot ser suplida amb la recopilació de molts exemples funcionals per tal de poder realitzar enginyeria inversa, fet que millorarà la corba d'aprenentatge. El desconeixement de la eficàcia de les eines exigirà que a l'inici de cada fase s'utilitzi més d'un software, per tal de comprovar quin s'ajustarà millor a les necessitats del moment.

Relacionat amb el producte, el riscos més grans són:

- Desconeixement del “món” en el qual es desenvolupa la funcionalitat del projecte.
- La dificultat d'interactuar amb el sistema per part dels usuaris un cop realitzat.

El programari s'utilitzarà en l'entorn d'un gimnàs, amb persones alienes a l'àmbit de la informàtica interactuant amb dades que el programador no ha vist mai (les qualitats de les quals desconeix, com en l'exemple de les dades de federació de *Taekwondo* es desconeixia la seva data de caducitat, fet que va obligar a re-dissenyar algunes taules un cop ja implementades). Per això en totes les fases del procés s'ha de preguntar i provar cada funcionalitat amb els administradors del gimnàs, resultant en una minimització dels problemes.

Relacionat amb el servei, el riscos més grans són:

- Pèrdua de dades per fallades mecàniques.
- Sobrepassada física de les capacitats del servidor.

Tot software s'executa en una màquina, la qual pot fallar en qualsevol moment. Per evitar pèrdues de dades, es projecta realitzar còpies de la base de dades un cop a la setmana. El segon risc també es possible, ja que el serveis que ofereixen els servidors contractables poden, en un moment donat, no ser suficient per la càrrega que rebin. Exceptuant un atac de denegació de servei o algun altre atemptat contra la web, es projecta contractar un servei que oferta més tràfic mensual i ample de banda del necessari, ja que la diferència de preus es tolerable.

Relacionat amb la comunitat, el riscos més grans són:

- Associació de malfactors en el portal.
- Atac contra el servidor i els serveis que ofereix.

Aquesta aplicació web permet el registre d'usuaris. Avui en dia, hi ha multitud de *malbots* (màquines que realitzen tasques no del tot legals, i directament il·legals), els més senzills dels quals recapten correus electrònics dels apartat d'inscrits per enviar-los publicitat enganyosa o estafes. S'ha de protegir les dades dels clients, impedit l'accés d'aquests falsos usuaris de recaptació de dades. També hi pot haver accés de menors, per tant els missatges privats entre usuaris ha de poder-se veure a nivell administratiu. A més, aquesta contindrà dades administratives, fet que la converteix en possible objectiu dels anomenats bucaners d'aigua dolça (*hackers* d'anar per casa que volen enganyar als administradors, per exemple que es conti el mes actual com a pagat) així com de grups que busquin dades bancàries. Les dades sensibles es protegiran, vigilant a més les activitats dels usuaris en el portal.

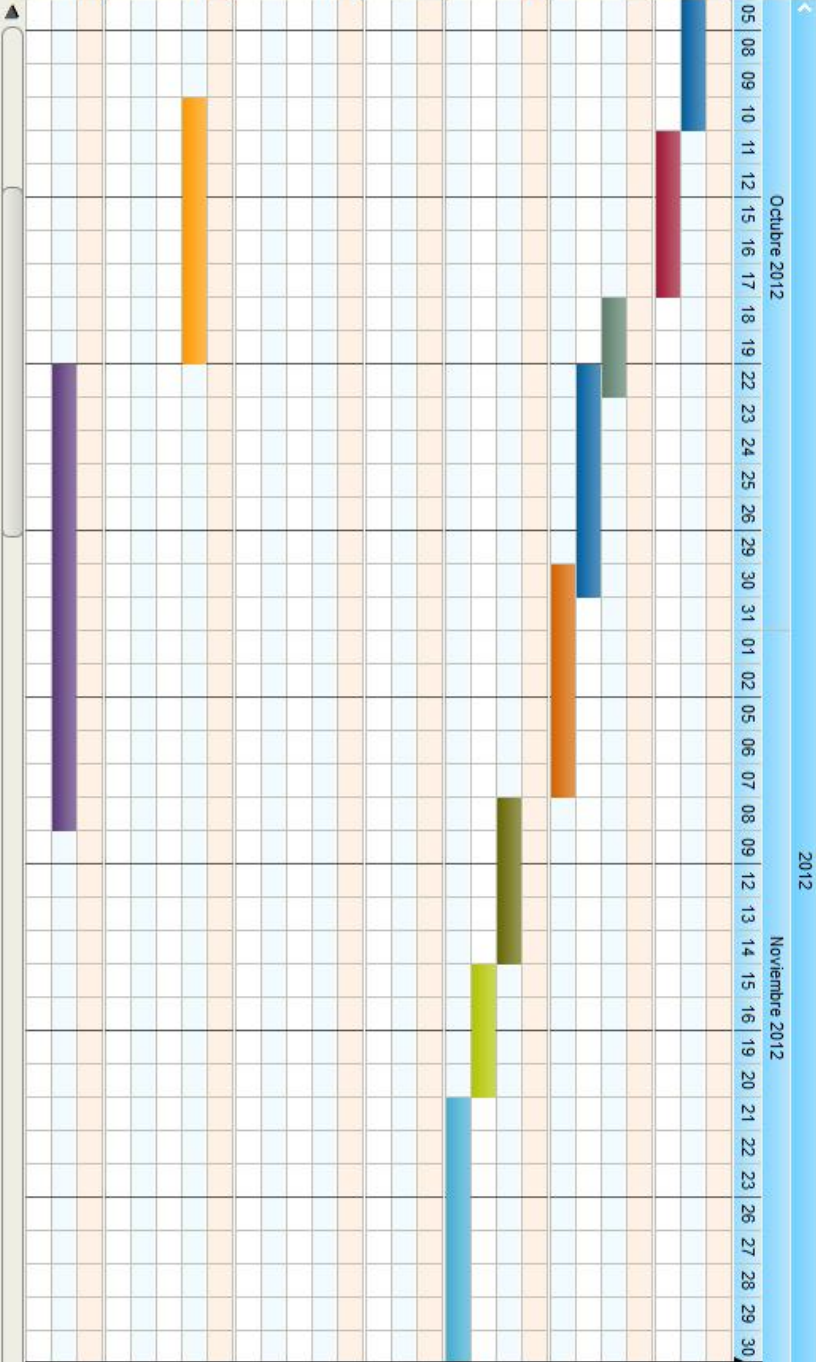
La defensa del servidor s'ha de tenir en compte a l'hora de contractar-lo, i realitzar còpies de seguretat cada setmana per si d'alguna manera es perden.

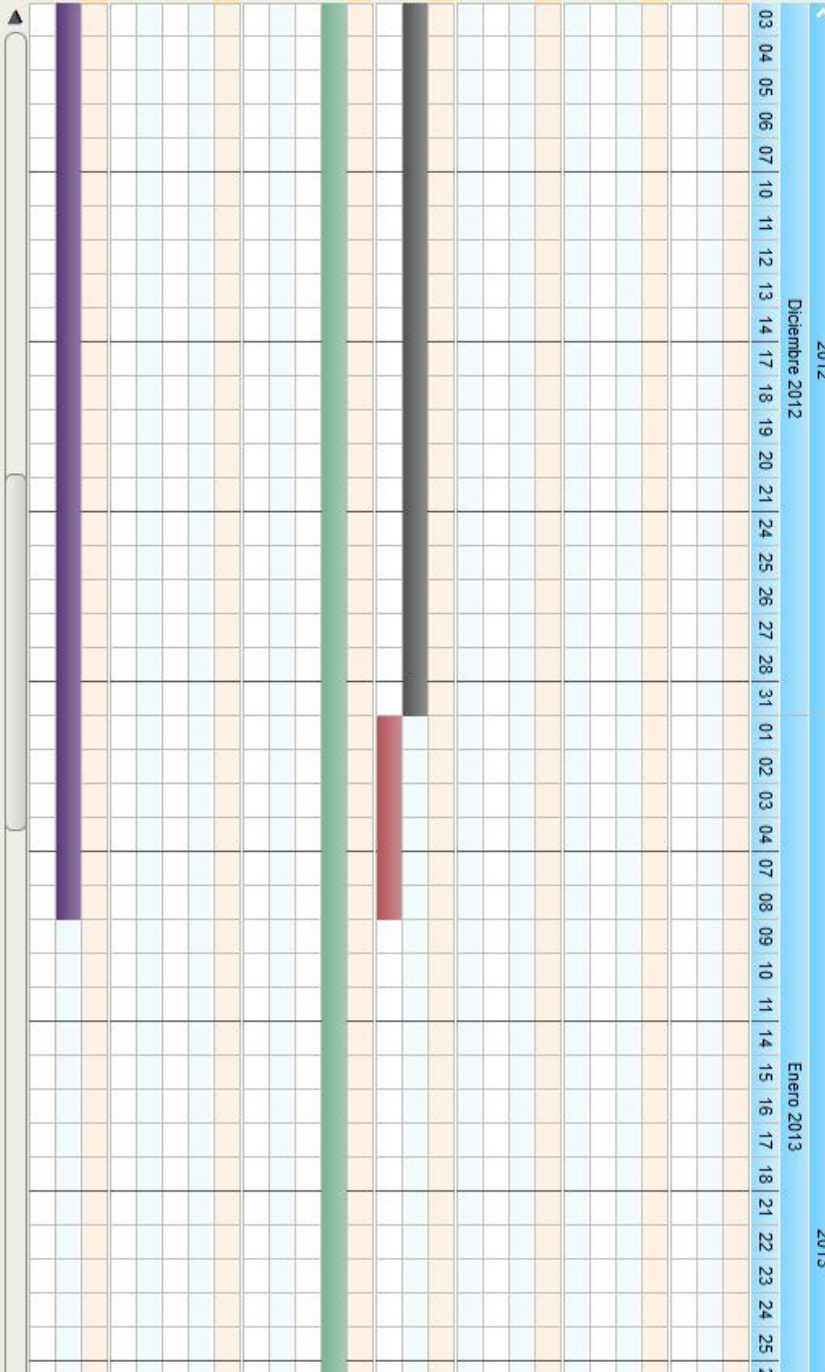
A nivell del client, no suposo cap risc, ja que en principi no hi ha d'haver cap canvi de requisits durant el projecte (s'han definit suficientment bé) i no hi ha pressió en el pla d'entrega en cas d'endarrerir-se, ja que el client

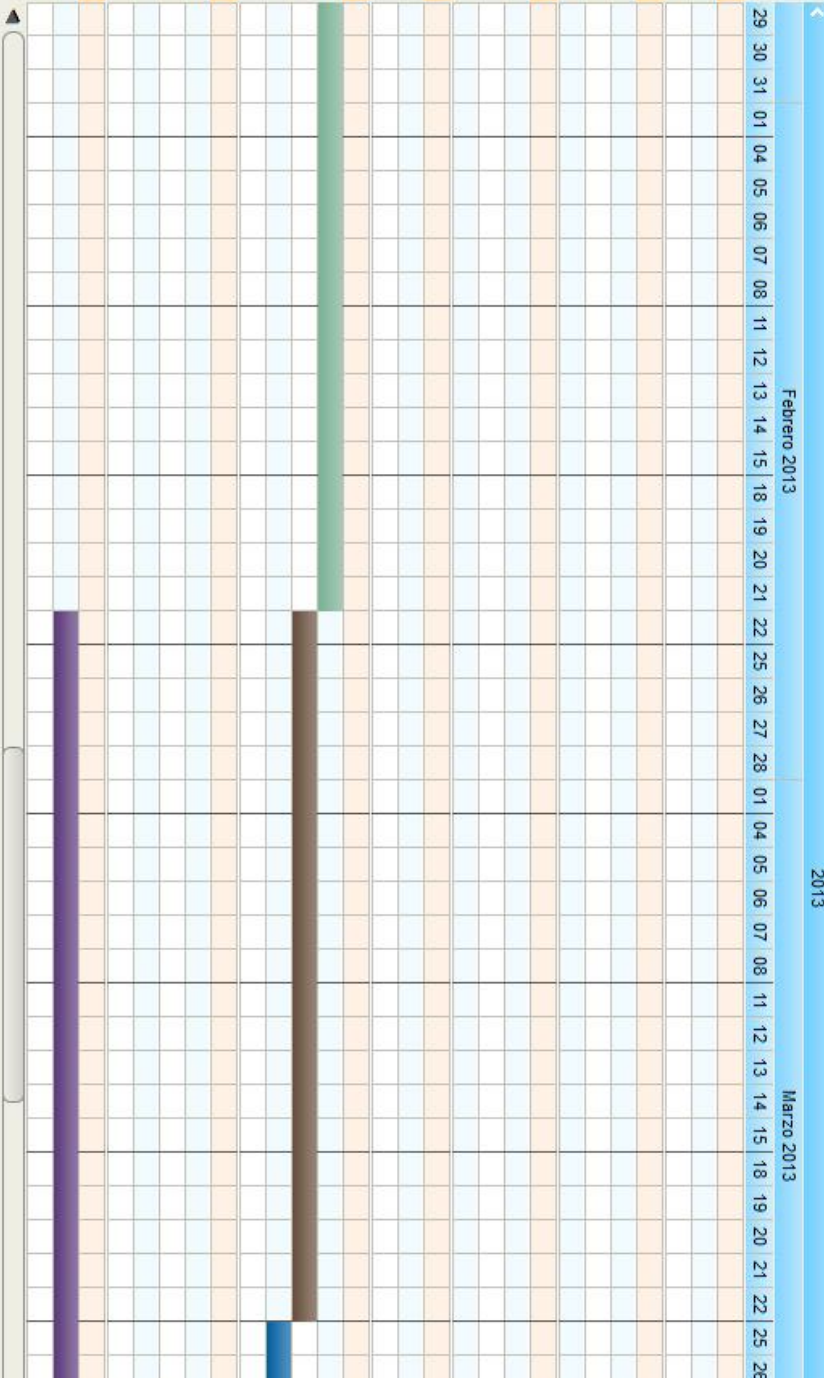
busca una millora del sistema que té funcionant, no un canvi urgent.

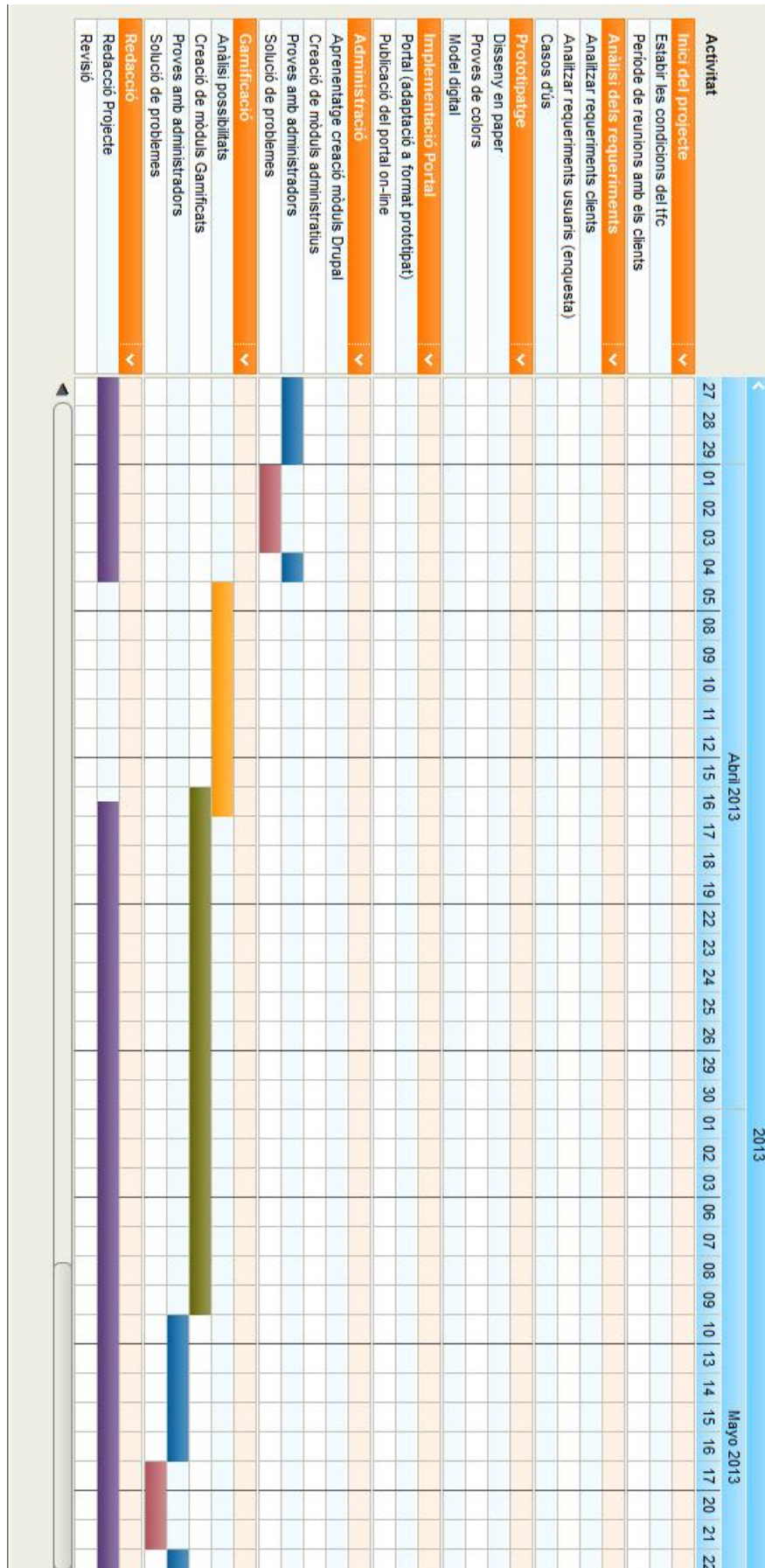
1.6 Planificació:

El projecte s'inicia amb la planificació del temps que trigarà a assolir-se cadascuna de les parts d'aquet treball. Es realitza un diagrama de Gantt a través de l'eina online (<https://www.tomsplanner.es>), en que es tracta aquest tema de manera visual per tal de comparar el temps projectat amb el real.









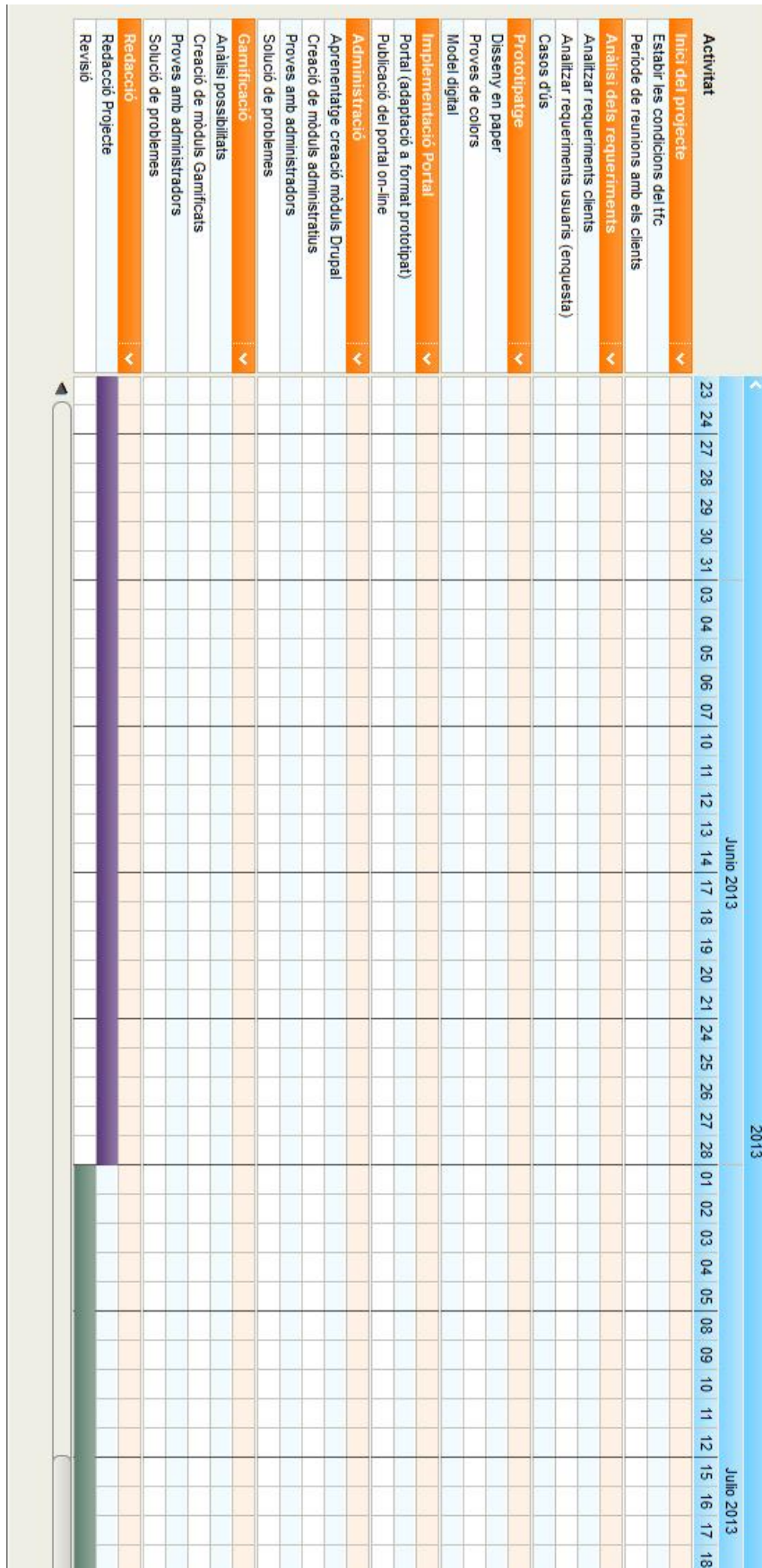


Diagrama 1: Diagrama de Gantt

El temps aproximat emprat per a la realització d'aquest projecte ha estat des del dia 5 d'Octubre de 2012 fins al 20 d'agost de 2013, quasi onze mesos. 286 dies projectats per dur-se a terme, tot i que finalment va requerir 319. Els segments de temps s'han mantingut, però el projecte s'ha endarrerit vora un mes i mig degut a un període de malaltia de l'autor del treball, just a continuació de la publicació del portal en línia en que va tenir que guardar llit tot el mes.

Havent vist el resultat final de la primera fase però, des del centre s'han mostrat molt satisfets i amb moltes ganes d'activar les competicions gamificades com a novetat després de la parada de vacances de l'estiu de 2013.

1.7 Costos:

Aquest projecte requereix de l'ús d'un servidor i en quant a costos el millor en relació qualitat/preu sembla ser el que ofereix l'empresa espanyola *Nominalia*. El paquet que anomenen *Premium hosting* té un preu de 12,42 € al més (que sumant-li el 21% d'IVA puja a 15,02 € mensuals).

En quant a la mà d'obra, tot i que es va projectar com a necessari 286 dies per dur-se a terme, finalment va requerir 319, amb un sol enginyer. El preu orientatiu d'aquesta tasca (encara que el treball s'ha realitzat sense cap cost pel centre com a part del treball de final de carrera de la llicenciatura) s'ha calculat en base a uns guanys del 20% a l'equació: (Sou per hora x dies x 4 hores). El sou d'un programador *php* a Espanya es troba entre 35 i 50 euros, així que es suposa un sou de 35 euros l'hora. Per tant el preu orientatiu és de: $(35 * 286 * 4) * 1.2 = 48.048$ euros, entre portal, software administratiu, i 20% del producte per al manteniment i el suport anual a noves funcionalitats.

Això es clar, inclou els processos d'aprenentatge. Per tant, el temps real de tornar a realitzar la tasca, coneixent-la millor després d'haver fet el projecte, s'ha calculat en:

- Establir les condicions amb el clients: 2 hores. (*No facturables, ja que en aquesta fase s'està venent el producte.*)

- Proves als usuaris: 2 sessions de 2 hores.
- Establir cassos d'ús: 2 hores.
- Realitzar el prototipatge i les diferents proves: 2 sessions de 6 hores.
- Implementació d'un portal web: 5 sessions de 4 hores.
- Personalització dels mòduls necessaris: 5 sessions 6 hores diàries.
- Imprevistos: 10 hores.

Això fa un total de $4+2+12+20+30+10 = 78$ hores. Seguint la mateixa norma que en el càlcul anterior, el preu orientatiu de realitzar un projecte amb l'experiència acumulada és de: $(35*78)*1.2 = 3276$ euros, que si afegim el cost del servidor el primer any s'obté un total de **3456,24 euros**.

Bloc 2:

Estat

Per a dur a terme aquest projecte, s'han utilitzat una sèrie de tecnologies diferents. En aquest segon bloc es fa referència als estudis realitzats als usuaris a qui va destinat el servei, el qual ens han permès establir les tecnologies amb les que es realitzarà el projecte.

2.1 Metodologia:

Seguint el model *MPIu+a* (acrònim de *Model de Procés de l'Enginyeria de la usabilitat i de l'accessibilitat*, que intenta compaginar el model de desenvolupament de sistemes interactius amb els principis bàsics de l'Enginyeria de la Usabilitat; extret de *Diseño de sistemas interactivos centrados en el usuario*, editorial UOC), s'han establert tres etapes que determinen la fase de desenvolupament: Disseny, prototipatge i avaluació (Figura 4).

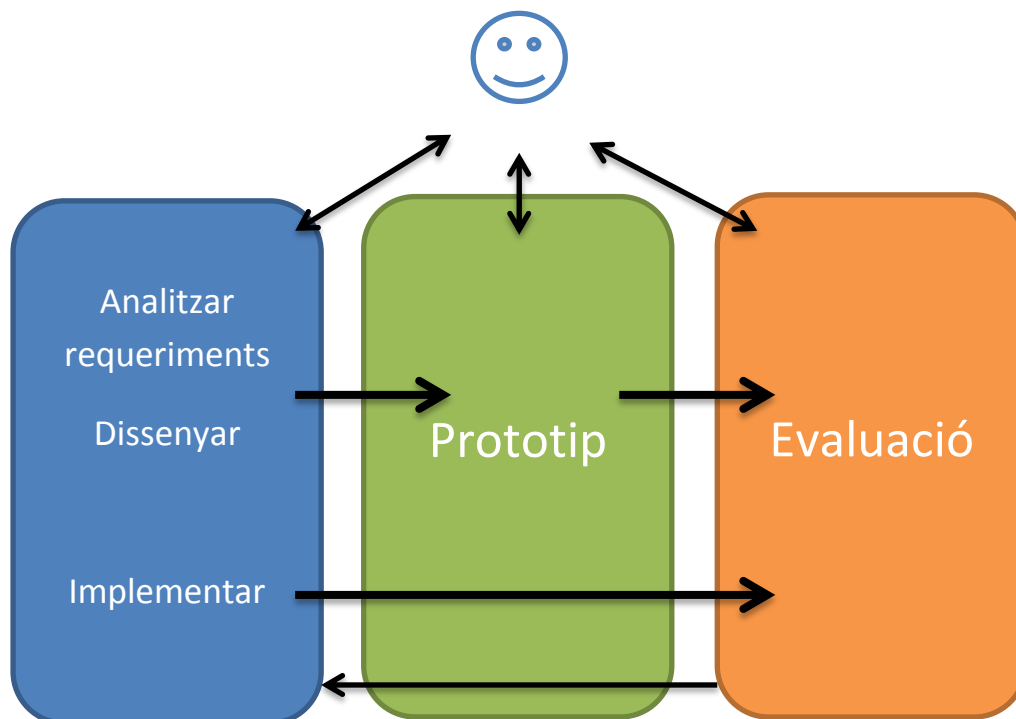


Figura 4: Fases de desenvolupament

L'usuari final s'inclou en el desenvolupament (o almenys una petita mostra

d'ells) avaluant el resultat en cadascuna de les etapes.

2.1.1 Requeriments:

La primera etapa es realitza abans de cap disseny, ni visual ni esquemàtic. Cada tipus de centre i usuari compta amb una sèrie de necessitats que s'han de cobrir. Es procedeix a analitzar als implicats.

2.1.1.1 Tipus de centre:

En aquesta ciutat hi ha tres tipus de gimnasos, cadascun enfocat a un tipus de públic clarament diferenciat.

El primer tipus es el dels gimnasos “industrials”, generalment locals de moda amb una gestió descentralitzada e impersonal, enfocada a gent jove en general. Aquests negocis compten amb grans instal·lacions i multitud de monitors especialitzats que realitzen el seguiment dels clients. Són els més informatitzats degut a la quantia de gent que maneguen, tot i que es redueix a tasques administratives. El client només té la opció de fer reserves d'espai telemàticament. Aquest gènere té com a màxima expressió en la ciutat l'*Ekke*.

El segon tipus és el dels gimnasos familiars de mida mitjana. Aquest es el sector al que va enfocat aquest treball. Es tracta de llocs amb una atenció personalitzada (i per tant necessàriament amb un volum inferior), on s'agrupen diferents tipus de perfils d'usuari, des de la recuperació mèdica fins l'alta competició. Les seves instal·lacions són més petites, però no de menys qualitat. De la mateixa forma, els seus responsables també són gent amb estudis especialitzats. Realitzat un anàlisis dels centres d'aquest segment, es constata que no tenen cap sistema de dades integrat, funcionant tots encara amb la tècnica de les “fitxes” emplenades a mà o en una base de dades molt bàsica (fulla *Excel*) gestionades des d'una mateixa màquina, sense interacció amb la Xarxa. A més, les seves pàgines web no contenen elements interactius pel públic, ni material que variï amb el temps. En són exemple el *Gimnàs Bordeta*, l'*Olympus* i el centre *Do San Lee*.

El tercer segment el conformen els gimnasos petits enfocats únicament al

fitness i/o culturisme, que no participen en competicions ni compten amb personal especialitzat.

Els requeriments del centre:

- **Promocional.** Informació per als potencials clients (instal·lacions, serveis que ofereixen, localització del centre, telèfon...). Promoció de tot el que s'ofereix i de la filosofia del mateix.
- **Evitar la tasa d'abandonament.** Mantenint l'interès dels usuaris.
- **Estimular la pràctica en grup de l'esport.** Amb eines desenvolupades per promoure aquesta tasca.
- **Fomentar l'autosuperació.** Mitjançant la consecució de reptes personals.

2.1.1.2 Usuaris:

S'estableixen dos tipus d'usuari, amb requeriments diametralment oposats que s'han d'integrar en l'ús d'aquest projecte: Els clients i els administradors.

Els requeriments dels administradors són pràctics:

- **Software d'administració de clients** integrat al portal, per:
 - Gestionar **comptes** de clients.
 - Controlar **assistències** diàries.
 - Controlar estatus de **pagament** de cada client que assisteix.
 - Tenir **fàcil accés a les dades** d'usuari per minimitzar el temps de recerca en cas de ser necessari (com les dades de la federació de taekwondo, que es requerida per l'asseguradora en cas d'accident).
 - Aquest ha d'estar disponible i **usar-se bé en un dispositiu mòbil**.

Mentre que els usuaris requereixen, a priori:

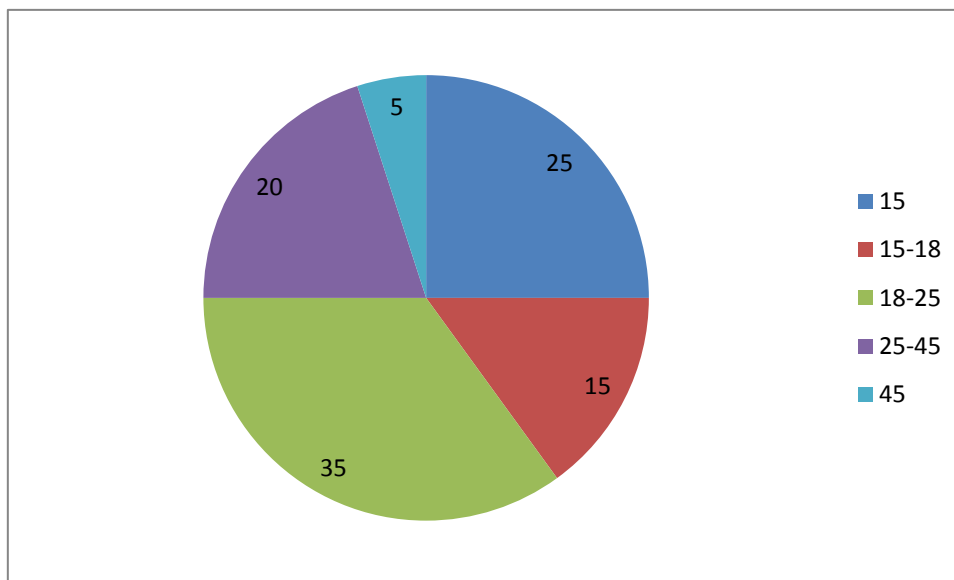
- **Portal accessible des de qualsevol navegador i sistema operatiu**, per

tant adaptat a múltiples resolucions de pantalla. Tot i ser autoescalable, es proposa que canviï l'aspecte segons l'ample de resolució sigui **inferior a 460 píxels** (pantalla mòbil antic), **entre 460 i 900 píxels** (pantalla de *tablet* o mòbil nou) i **més gran de 900 píxels** (pantalla d'ordinador, generalment).

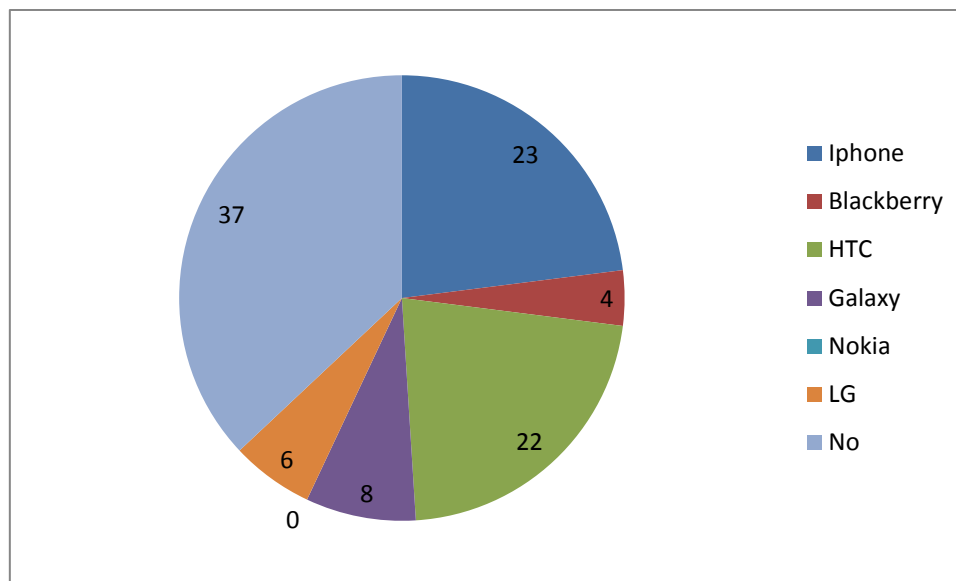
- **Portal amigable** on trobar informació a nivell:

- **Informativa.** Informació del gimnàs (com arribar-hi, telèfons, qui són...), amb un aspecte agradable e intuïtiu.
- **Activitats que es realitzen normalment** i els seus horaris.
- **Activitats extraordinàries** que es realitzen i els horaris en que ho fan.

Per tal d'avaluar si els usuaris tenen més requeriments, es realitza l'enquesta que es troba en l'**Annex A** (cinquanta en català i cinquanta més en castellà). Els resultats es troben en el directori *0-ResultatsEnquesta* del DVD adjunt a aquest treball.



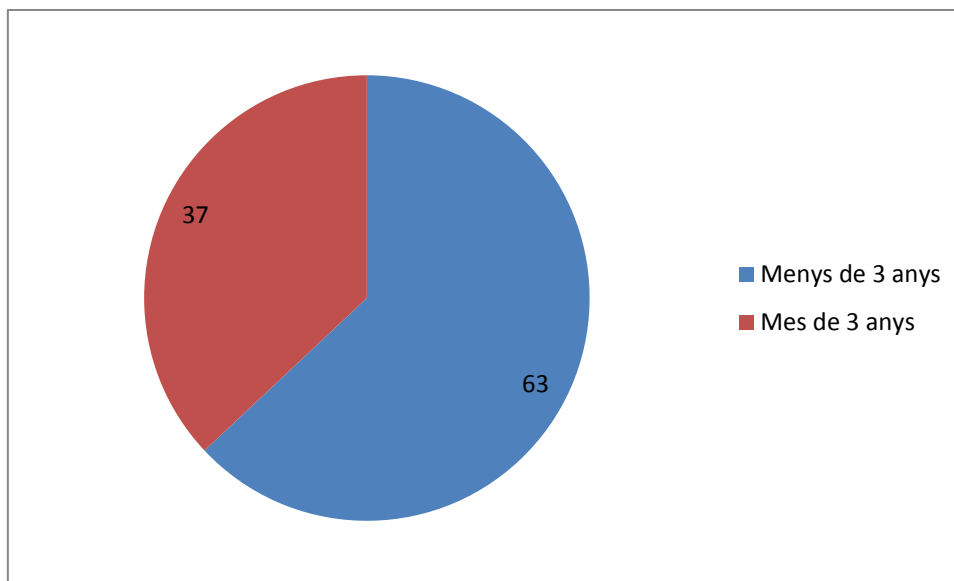
gràfica 1: Edat.



gràfica 2: Tenen internet als mòbils?.

Com es pot veure, hi ha una població molt jove (gràfica 1), amb accés a internet des de tots els llocs en un 63% (cosa que segons els administradors es un problema degut a que perden l'atenció) i mòbils relativament potents amb els que fer-ho, gràfica 2. La Xarxa social amb més vida per ells es *Facebook* amb diferència (61%), tot i que *Twitter* arrasa entre el públic més jove (27% del total). Pocs d'aquests accedeixen a la pagina de *fb* (28%) del gimnàs, que no te compte de *twitter*.

Mantenir la forma i preparar competicions és el primer motiu per assistir al centre, però la realització de les activitats obté un percentatge alt també en les necessitats de la gent. Les preparacions per a proves d'accés no, tot i que probablement es degut a que durant la realització de l'enquesta no hi havia cap curs al respecte (el centre en realitza habitualment).



gràfica 3: *Fidelitat dels clients*

Sempre hi ha usuaris “de temporada” (que s’apunten en períodes d’estiu i de propòsit d’any nou), però el gimnàs té molt de client habitual de més de tres anys d’assistència, a més de joves clients que “han crescut” dins. Tot i això, degut a la seva secció infantil la tasa de nous inscrits cada any es manté.

Els usuaris també podrien estar interessats en:

- Interacció amb **xarxes socials**.
- Establir una línia de **comunicació amb el centre**, per a consultes de tota mena.
- **Competicions virtuals** amb les que donar a conèixer i posar en comú les seves fites i estat. Amb això s’incrementa tant el valor dels seus esforços com l’*experiència* que viuen.

2.1.1.3 Necessitats especials:

Tot i ser joves de mitjana, alguns usuaris poden tenir alguna necessitat especial. Les discapacitats (tant permanents com temporals) que poden afectar a l’ús del portal són bàsicament les visuals, les motrius i les cognitives. Per permetre l’accessibilitat als grups afectats per una o més d’aquestes necessitats s’han adoptat un seguit de mesures (que serveixen

per poden desenvolupar la fase de disseny):

- **Discapacitats visuals:**

La ceguera no s'ha contemplat (tot i que amb els dispositius adequats l'assistència l'hauria de donar el navegador, sempre i quant estiguin ben etiquetats els elements de la web), però si la visió reduïda i els problemes en la visualització dels colors.

Per a millorar la seva visualització (així també se'n beneficien els usuaris sense afectació) s'han escollit mides grans de text (que a més no es troben prefixades en els *css*, adaptant-se a mida de text de l'usuari), amb distàncies entre paràgrafs espaiades i botons grans que ressalten a la vista.

S'ha assegurat un alt contrast entre els diferents colors que es poden trobar a la web. L'esquema de colors seleccionat s'ha escollit amb l'assistent web *colorshemedesigner.com*, que compta amb un assistent geomètric que ens proposa colors oposats i suficientment distanciat com per ser diferenciades en cas de ceguera al color i dificultats amb els mateixos.



Figura 5: Esquema de colors del portal

A més, es comprova constantment durant el procés que no hi hagi afectació degut a cap malaltia visual amb un altre assistent web, en aquest cas <http://www.dasplankton.de/ContrastA/>. Aquest comprova les combinacions de color de contrast i mostra els resultats en funció de les WCAG 2.0 (Relació de luminància), així com els resultats d'acord a les pautes d'accessibilitat WCAG 1.0, més antigues (diferència de brillantor i color).

A més mostra els resultats per a la visió normal i simula els colors per als diferents tipus de deficiències visuals de color en les persones.

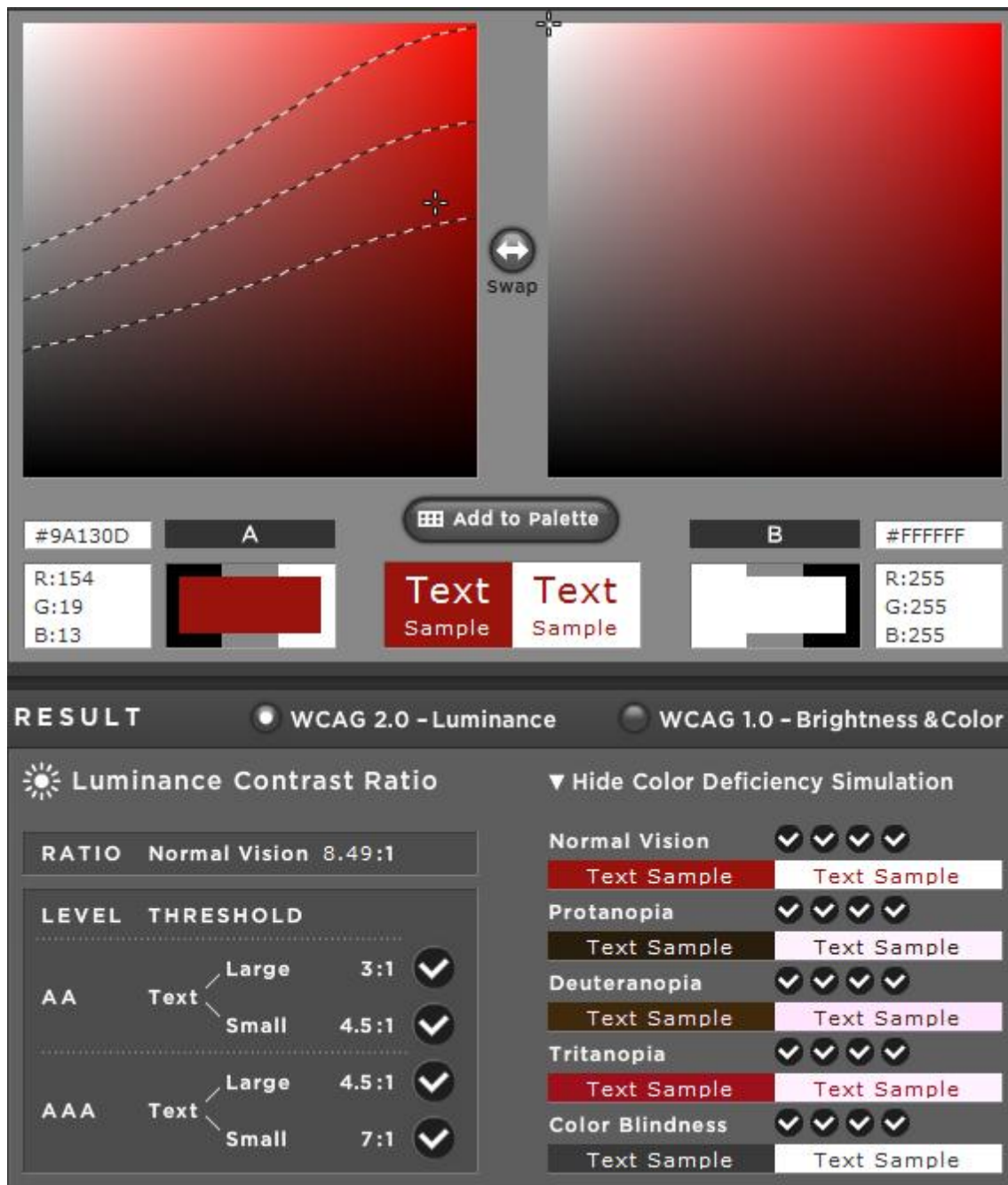


Figura 6: Proves de visió

• Discapacitats motrius:

Alguns usuaris podrien presentar problemes amb la mobilitat, i per tant tenir dificultats per interactuar amb el sistema. Per minimitzar aquestes problemàtiques, la mida dels botons dels enllaços és variable depenent de la resolució però sempre suficientment grans com per poder interactuar

sense cap problema. Per als usuaris de *tablet* o mòbil (problema típic de dits massa grans), s'han adaptat uns grans botons de menú principal per a més comoditat. A més no és necessari l'ús de teclat (físic) i ratolí per interactuar-hi.

- **Discapacitats cognitius:**

Pels usuaris en general, la navegació simplificada millora la experiència, però pels usuaris que presenten problemes de percepció o de comprensió (ja sigui per malaltia o per trobar-se més al extrem del que es considera “normal” avui en dia) és bàsic que l'estructura del portal sigui el mes simple possible: Botons clarament diferenciats, amb textos mínims que marquin exactament el que fan.

2.2 Tecnologies:

Les tecnologies amb les que s'ha interactuat per tal de produir aquest treball es troben definides pels estudis de l'apartat anterior, resolvent que es necessitaran els següents llenguatges i programes.

2.2.1 Llenguatges:

Aquest projecte busca la creació d'una aplicació web. Per tant, es requereix l'ús de *HTML* (llenguatge base que interpreten els navegadors web). A més inclourà scripts per crear continguts dinàmics, realitzats en *php* i *Javascript*, a partir d'elements guardats en una base de dades *SQL*. La representació visual es separarà de l'etiquetatge *html* utilitzant fulles d'estil (*css*).

HTML: Acrònim d'*Hyper Text Markup Language*, llenguatge de marcat que basa el seu funcionament en l'ús d'etiquetes per estructurar textos representar-los posteriorment en una interfície web. Al interpretar les etiquetes, sense compilar, qualsevol errada es mostra tal i com s'ha fet, sense missatges d'error.

El problema radica en que *html* només genera continguts estàtics, per generar continguts dinàmics es necessària la execució d'algun programa en el servidor que retorni continguts en *html* construïts al moment (a partir de una programació determinada).

CSS: *Cascading Style Sheets* és un llenguatge basat en fulls d'estil que descriu el format i l'aspecte d'un document *HTML*. S'utilitza per separar tant conceptualment com físicament el codi amb el que s'estructura la representació de les dades que mostrarà la web, del codi de la representació visual. A més ajuda a mantenir el mateix aspecte corporatiu en tota la pàgina.

PHP: Descriuint el seu funcionament, acrònim recursiu de *Hipertext Preprocessor*, tot i que originalment eren les sigles de *Personal Home Page Tools*. Llenguatge de programació interpretat utilitzat per a generar pàgines web de forma dinàmica. És lliure, permet aplicar tècniques de programació orientada a objectes i disposa d'una biblioteca nativa de

funcions summament àmplia i inclosa. S'executa en el servidor, on genera una pagina nova a partir de la petició que ha rebut, per aquest motiu al navegador del client ja hi arriba la pàgina en format *HTML*, no podent visualitzar-ne el codi *php*. Permet que múltiples pàgines comparteixin un format comú i redueix la complexitat i repetició en el contingut estructural, amb una pàgina de creació de continguts que construeix els elements amb la informació que te guardada en una base de dades *SQL*.

Javascript: Malgrat el seu nom, *JavaScript* no deriva del llenguatge de programació Java, però tots dos comparteixen una sintaxi similar. S'utilitza sobretot per afectar en temps real al contingut de la web, movent elements o executant instruccions a partir de les accions de l'usuari (canvis en el codi que *HTML* o *php* no poden realitzar una vegada establerts).

MySQL: Sistema de gestió de bases de dades relacional, multi-fil i multiusuari, que usa el llenguatge *SQL*. Compta amb una gran velocitat d'execució de consultes i (era, fins la versió 4) suportat de forma nativa per *php* en l'elaboració d'aplicacions web, en l'entorn del programari lliure. Es pot fer ús de MySQL en aplicacions de tota de forma lliure i gratuïta sota les condicions de la llicència GPL, tot i que per realitzar un producte privatiu cal adquirir una llicència d'ús específica per a aquest propòsit.

2.2.2 Programari utilitzat pel desenvolupament:

Per la realització d'aquest projecte, s'ha requerit l'ús d'alguns programes per diverses tasques. *Photoshop* i *Axure* per al desenvolupament del prototip digital; *Drupal 7* per a la gestió del contingut online, *Notepad++* per a l'edició dels fitxers de configuració i *Filezilla* per a interactuar amb el servidor via *ftp*; *XAMPP* per a la creació d'un model funcional en local; *Apache 2.2.9*, *phpMyAdmin 3.3.7* i *PHP 5.2.6-1* per l'execució online del model local.

Photoshop CS4: Editor de gràfics desenvolupat per *Adobe Systems*. Actual líder de mercat per a mapes de bits comercials i manipulació d'imatges, complint l'estàndard industrial en moltes feines relacionades amb l'ús d'elements visuals.

Inkscape: Editor de gràfics *open source*. Ideal per manipular els gràfics vectorials (en format *svg*) amb els que s'han dissenyat les medalles de gamificació.

Axure RP: Eina de prototipatge orientada a aplicacions web. Ofereix capacitats d'emplaçament *drag and drop*, reescalat i assignació de formats sense tenir que interactuar amb codi *html*, de manera visual i esquemàtica. A partir d'aquests diagrames, genera un prototip en *HTML* pla que, tot i no ser gens eficient, sí que es funcional, donant una idea de l'aspecte general que ha de tenir el projecte.

Drupal 7: Sistema de gestió de continguts per crear aplicacions web. La seva arquitectura modular, la seva API i la capacitat de configuració, permeten creat tot tipus de portals optimitzant el temps de desenvolupament. El fet que gairebé tots els aspectes del comportament de *Drupal* poden ser interceptats i transformats a través dels mecanismes de mòduls i tema, ha portat a afirmar que Drupal no és només un sistema de gestió de contingut (*CMS*), sinó un *Framework* de gestió de continguts (*CMF*) capaç de ser reconstruït per satisfer les necessitats i els requisits funcionals específics. Realitza totes les funcions estàndard d'un portal basat en web. Els visitants poden veure la informació publicada al lloc, navegar pels menús i veure pàgines. També poden crear comptes i deixar comentaris. Els administradors poden gestionar el lloc, configurar-lo i controlar els nivells de permisos dels usuaris; crear continguts, obtenir una vista prèvia i, a continuació, publicar, una vegada tot estigui llest. El contingut pot ser sindicat en *RSS*, on els lectors de *feeds* poden recollir els nous articles que es publiquen. Fins i tot l'aparença de la pàgina es pot canviar fàcilment sense necessitat d'interactuar amb codi.

Notepad++: editor de text i en particular de codi font per a Windows. El seu objectiu és ser un editor lleuger i robust per a una varietat de llenguatges de programació i scripting. Un avantatge al predefinit *Notepad* de Windows, és que és compatible amb l'edició de pestanyes, que permet treballar amb diversos arxius oberts alhora. És compatible amb ressaltat de sintaxi i accepta més de 50 llenguatges d'scripting, programació i etiquetes. Tot i això, la seva gran funcionalitat consisteix en que es capaç de guardar fitxers en format *UTF8* sense marca *BOM* (a diferència del predefinit *Notepad* de Windows), el qual ens permet que *AJAX* accepti els caràcters

en català sense causar cap malfuncionament.

XAMPP: Paquet de programari lliure que conté el un servidor *HTTP Apache*, bases de dades de *MySQL* i les eines necessàries per utilitzar *PHP* i *Perl*. El programa simula un servidor web en local, d'ús fàcil, capaç de servir pàgines dinàmiques, de manera que els programadors web poden provar els seus programes sense connexió a internet.

Apache: Servidor HTTP de codi obert multiplataforma desenvolupat per l'*Apache Software Foundation*. El seu nom ve de que als seus inicis consistia en una sèrie de mòduls que s'acoplaven, conformant un *patchy server* (servidor apedaçat). L'arquitectura del servidor Apache és molt modular; alguns dels seus mòduls donen compatibilitat per a *perl*, *php* i bases de dades entre d'altres.

PHP MyAdmin: Eina d'administració escrita en *php* amb la funció de gestionar *MySQL* a través d'interfície web, mitjançant navegador. Pot crear i eliminar Bases de dades, crear, eliminar i alterar taules, esborrar, editar i afegir camps, executar qualsevol sentència SQL, administrar claus en camps, administrar privilegis, exportar dades en diversos formats i està disponible en 50 idiomes. Es distribueix sota la llicència GPL.

PHP: Intèrpret per als scripts d'instruccions programat mitjançant el llenguatge homònim. Multiplataforma i distribuït sota llicència Open Source, és un sistema molt versàtil.

2.2.3 Característiques de la plataforma d'execució:

Aquest projecte es desenvolupa com a una plataforma online. Degut a que el centre no té personal especialitzat en informàtica es resol de contractar el servei de *hosting* i de servidors a *Nominalia* (Empresa espanyola de bona relació qualitat/preu). Per tant, les característiques de la màquina física no s'especifiquen, ja que el servei en proves anteriors ha semblat ser suficient. Al executar-se sobre *Drupal*, sempre es pot migrar una còpia a un altre server en menys de 15 minuts (comprovat).

Bloc 3:

Disseny

Amb els requeriments de l'apartat anterior, es procedeix a realitzar el disseny del portal web. Amb la conjunció d'aquests dos elements, es proposa un primer prototip en paper i un cop avaluat, es concreta en un prototip digital que servirà de guia per la posterior implementació del projecte.

3.1 Arquitectura de la informació:

La estructura i organització d'una aplicació és una part tant important com la mateixa implementació.

Per estructurar el menú principal es va recórrer a la tècnica de *card sorting*. Els administradors del gimnàs van escriure en targetes quins apartats hauria de tenir el portal i un grup d'usuaris van rebre les targetes per a que les ordenessin tal i com pensaven que s'havien de trobar.

Les diferents versions de l'estructura del títol i els botons de la pàgina principal van ser analitzades per un professional de la usabilitat, que va proposar canvis per al model definitiu.

A més, en fases posteriors del treball, es van reordenar els camps dels formularis i l'estructura de les pàgines a petició d'alguns usuaris, per tal de millorar la lògica de l'estructura.

3.1.1 Card Sorting:

Aquesta tècnica es basa en l'ordenació de targetes les qual representen els diferents temes que conformen el portal. En el disc que s'inclou juntament amb aquest projecte, es poden trobar les fotografies en la taula on es van realitzar les diferents proves de *card sorting*, fins arribar a la definitiva. Dos socis van establir els apartats. Aquests van ser classificats per tres usuaris, que van proposar el model següent:

Qui Som

- Contacte

Horaris

Campionats

- En Curs
- Històric

Instal·lacions

- Sala Fitness
- Tatami
- Vestidors
- Jacuzzi
- Sauna
- Pista Squash
- Fotos
- Videos

Activitats

- Taekwondo
- Defensa Personal
- Aikido
- Kick Boxing
- Spinning
- Ioga
- Pilates
- Tai-chi
- Aeròbic
- Personal

3.1.2 Perfils d'usuaris:

El portal ha de tenir tres tipus d'usuaris diferents: Usuari Anònim, Usuari Registrat i Administrador. Per suposat cada usuari hereta tots els permisos del rol anterior. Les funcionalitats de l'aplicació segons el rol són:

Usuari Anònim (Aquest és un visitant ocasional de la web, que no sabem si es membre del gimnàs o podria voler ser-ho):

- Iniciar sessió (En cas d'estar registrat).
- Registrar-se en el portal (requereix aprovació d'administrador).
- Consultar informació promocional (telèfon, localització, horaris, activitats, serveis, activitats especials, notificacions i algunes competicions actives, a més de les xarxes socials).
- Utilitzar el formulari de contacte (per si de cas vol posar-se en contacte sense implicar-se tant d'entrada).

Usuari Registrat (Visitant que es client del gimnàs o que els administradors han acceptat com a tal). Diferenciarem dos rols per a aplicacions futures: Majors d'edat i menors, tot i que en la primera fase tots dos tindran les mateixes funcionalitats.

- Tancar sessió.
- Modificar les seves dades personals.
- Visualitzar la informació de totes les competicions actives.
- Enviar missatges a altres usuaris.

En cas de sol·licitar-ho expressament, s'habilitarà la funcionalitat experimental de **gamificació**:

- Participar (o no fer-ho) en alguna de les competicions internes actives.
- Consultar el seu perfil de gamificació, així com veure les

fites i nivells que han assolit en el transcurs de les competicions.

Administrador (com el seu nom indica, administra el portal). És l'usuari amb poders màxims sobre tots els elements del portal i els usuaris que es troben registrats

- Habilitar comptes d'usuari nous.
- Des habilitar comptes d'usuari.
- Crear nous continguts i modificar els vells.
- Modificar els menús.
- Inserir dades de clients (En una base de dades administrativa).
- Consultar dades de clients.
- Portar un control sobre les assistències dels clients.
- Afegir fites i associar-les als usuaris.
- Afegir dades de campionats interns.

3.1.3 Casos d'ús i especificacions:

En disseny de software, un cas d'ús és una tècnica per a comprovar els requisits per a que una de les funcionalitats es compleixi. Aquests mostren l'escenari tal i com el programari haurà de funcionar. Els diagrames corresponents a les funcionalitats del portal es troben a les figures 7, 8 i 9.

Algunes tasques no requereixen una explicació molt elaborada (com per exemple consultar les instal·lacions, que seria només clicar al boto corresponent), però per unes altres el sistema té més complexitat (sobretot en els apartats d'administració i gamificació). Per cada tasca complexa en cada escenari, s'ha fet una taula que mostra la interacció del usuari amb el sistema.

3.1.3.1 Casos d'ús Portal:

Tots els portals tenen una sèrie de funcionalitats bàsiques. En aquest cas, *Drupal 7* n'inclou moltes referents a la gestió i interacció d'usuaris.

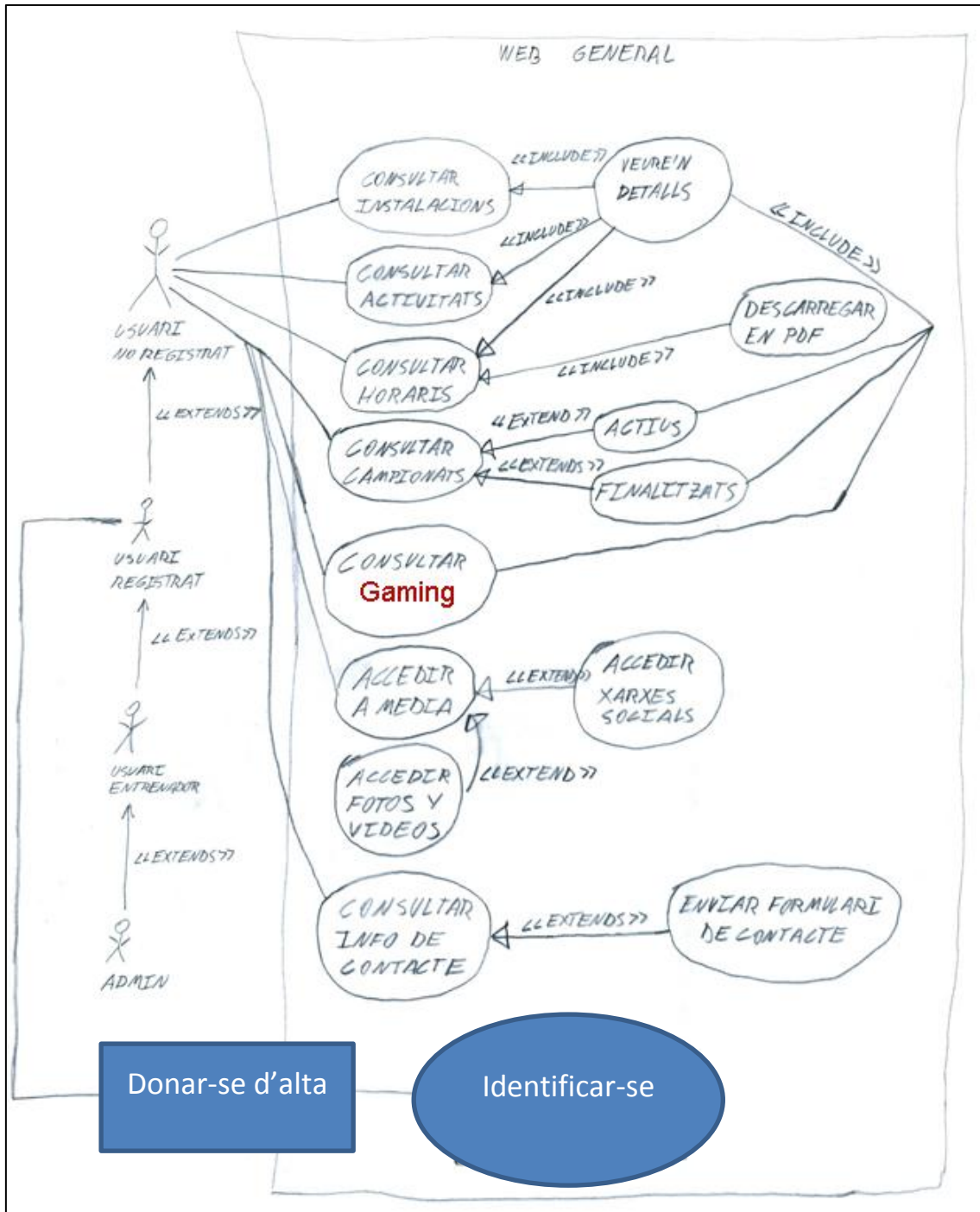


Figura 7: Casos d'ús portal

Aquest escenari (amb un usuari anònim) és el més simple i com a tal només interactua amb el sistema d'una manera més complexa en tres tasques:

donar-se d'alta, identificar-se i enviar un formulari de contacte.

Donar-se d'alta:

Un nou usuari que no té compte al portal vol formar-ne part.

Usuari	Sistema
1. Clica a dalt a la dreta, davall del nom d'usuari a "Crea un compte nou".	2. El sistema obre un formulari de registre on es demana el nom i la direcció de correu electrònic.
3. Introdueix un nom i correu electrònic vàlid.	4. El sistema envia un correu al usuari informant-lo de que esta pendent d'aprovació per part d'un administrador i un altre als administradors demanant aprovació. Els administradors donen l'aprovació i el sistema retorna un <i>mail</i> amb una contrasenya provisional.
5. Obre el <i>link</i> que li ha arribat al correu i segueix els passos de canvi a una contrasenya que només ell coneix.	6. El sistema ja el reconeix com a usuari vàlid.

Post condició: L'usuari es troba registrat al portal.

Identificar-se:

Un usuari actiu vol identificar-se en el portal.

Usuari	Sistema
1. Emplena nom i contrasenya als espais de dalt a la dreta i clica al boto "entra" o polsa la tecla <i>Intro</i> .	2. El sistema comprova la contrasenya i el nom d'usuari, i l'accepta.

Post condició: L'usuari es retornat a la pàgina principal del portal, ja identificat. A dalt a la dreta en comptes dels camps d'entrada surt el seu nom de perfil.

Excepcions:

2. Si el sistema no reconeix l'usuari o la contrasenya es errònia, informa al usuari amb un missatge per pantalla i li pregunta si s'ha oblidat de la contrasenya. En cas afirmatiu, envia un missatge a la direcció de correu que li especifiques (sempre i quant sigui el mateix correu amb el que l'usuari es ca registrar a la pàgina), tornant al punt 4 de l'apartat anterior (donar-se d'alta).

Enviar formulari de contacte:

Un usuari vol utilitzar el formulari de contacte integrat per posar-se en contacte amb els administradors.

Usuari	Sistema
1. Clica al boto "contacta".	2. El sistema obre un formulari de contacte on es demana el nom, una adreça de correu vàlida, un títol resum de la consulta i la consulta que es vol realitzar.
3. Omple els camps obligatoris i selecciona Consultes gimnàs o Website feedback, per escollir al subjecte al que arribarà la qüestió i clica al botó enviar.	4. El sistema envia un correu al Subjecte que ha especificat l'usuari, informant-lo de que ha usar el formulari web, amb les dades que ha donat adjuntes.

Post condició: L'usuari es retornat a la pàgina principal del portal.

Excepcions:

2. Si l'usuari es troba registrat i identificat a la web, el servidor suposarà que el nom i el camp de correu electrònic són els del usuari registrat.

3.1.3.2 Casos d'ús Administrador:

Drupal 7 inclou la possibilitat de gestionar continguts i comptes des d'un apartat d'administració, tot i que el centre requereix d'una sèrie de funcionalitats específiques que tindran que ser desenvolupades per l'ocasió.

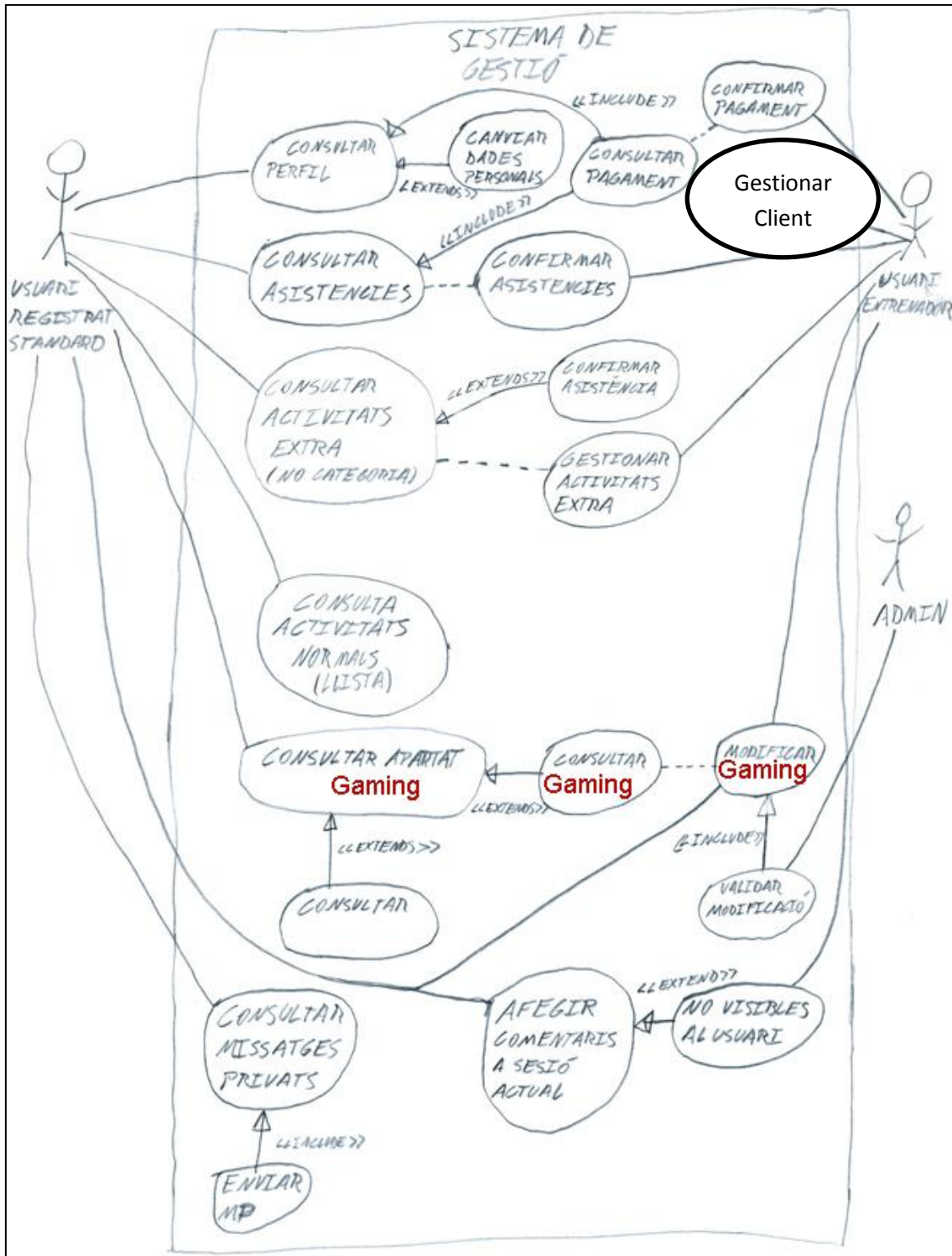


Figura 8: Casos d'ús Administrador

Aquest escenari és complex per al nivell administratiu. Totes les tasques possibles es troben organitzades en un menú sota l'etiqueta “Administrar”, només visibles per a l'administrador i a més amb seguretat extra inclosa. Els passos suposen que l'administrador ja es troba en el menú “Administrar”.

Introduir usuari en la base de dades:

Un nou usuari es entra en el sistema pel administrador.

Administrador	Sistema
1. Clica al boto “Introduir usuari en la base de dades” , sota l'apartat Administració del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris: nom, cognoms, NIF o NIE, data de naixement, adreça ciutat telèfons i mode de pagament.
3. Omple els camps obligatoris i dona al botó publicar.	4. El sistema compara la contrasenya i l'accepta. Comprova que aquest nom no estigui introduït en la base de dades i procedeix a encriptar les dades sensibles del mateix. Un cop fet mostra les dades que s'introduiran, les guarda en la taula de clients de la base de dades i informa al administrador de que s'han introduït amb èxit.

Post condició: Les dades de l'usuari són a la base de dades, amb alguns camps encriptats.

Excepcions:

3. Si l'usuari no ha entrat la contrasenya correcta, el sistema informa de que no es vàlida i retorna al punt 2.
3. Si ja existeix l'entrada, el sistema informa de que ja existeix i retorna al punt 2.
3. Si l'usuari no té assignat el rol d'administrador, no es fa cap consulta a la

base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

Federació de taekwondo – Introduir dades:

Un nou usuari de taekwondo es entrat en el sistema pel administrador.

Administrador	Sistema
1. Clica al boto “Federació de taekwondo: Introduir dades” , sota l’apartat Administració del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris: nom, cognoms, número de federació, data de caducitat de federació i cinturó actual.
3. Omple els camps obligatoris i dona al botó publicar.	4. El sistema compara la contrasenya i l’accepta. Comprova que aquest nom estigui introduït en la base de dades de clients. Un cop fet mostra les dades que s’introduiran, guarda la informació de la federació en la taula Taekwondo i les dades del cinturó en la taula Cinturo. informa al administrador de que s’ha actualitzat l’entrada amb èxit.

Post condició: Les dades de usuari es troben disponibles com a expansió de les que ja es troben a les altres taules.

Excepcions:

3. Si l’usuari no ha entrat la contrasenya correcta, el sistema informa de que no es vàlida i retorna al punt 2.

3. Si l’usuari no té assignat el rol d’administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

3. Si l’usuari no existeix en la taula client, no s’introdueixen els valors i es

mostra un missatge d'advertència. En aquest missatge s'inclouen els clients amb el mateix primer cognom que l'especificat, per a que al tornar a realitzar la consulta es vegi si s'havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Federació de taekwondo – actualitzar:

S'ha d'actualitzar les dades de federació (que caduquen) o les dades de cinturó d'un usuari de taekwondo.

Administrador	Sistema
1. Clica al boto “Federació de taekwondo: Actualitzar” cinturó o carnet, sota l'apartat Administració del menú.	2. El sistema obre un formulari on es demana els camps necessaris: nom, cognoms i element a actualitzar.
3. Omple els camps obligatoris i dona al botó publicar.	4. El sistema compara la contrasenya i l'accepta. Comprova que aquest nom estigui introduït en la base de dades de clients. Si hi es, comprova que també existeix a la base de dades de taekwondo. Mostra les dades que s'introduiran, guarda la informació de la federació en la taula Taekwondo i les dades del cinturó en la taula Cinturo. informa al administrador de que s'ha actualitzat l'entrada amb èxit.

Post condició: Les dades de usuari es troben disponibles com a expansió de les que ja es troben a les altres taules, sense haver tocat les anteriors, que queden com a historial.

Excepcions:

3. Si l'usuari no ha entrat la contrasenya correcta, el sistema informa de que no es vàlida i retorna al punt 2.

3. Si l'usuari no té assignat el rol d'administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant un “l'usuari no

existeix en la base de dades”, sense haver inclòs res a la base de dades.

3. Si l’usuari no existeix en la taula client i taekwondo, no s’introdueixen els valors i es mostra un missatge d’advertència. En aquest missatge s’inclouen els clients amb el mateix primer cognom que l’especificat, per a que al tornar a realitzar la consulta es vegi si s’havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Control Bancari – Introduir dades:

Un client acaba de realitzar un nou pagament i s’ha d’actualitzar a la base de dades.

Administrador	Sistema
1. Clica al boto “Control Bancari: Introduir dades” , sota l’apartat Administració del menú.	2. El sistema obre un formulari on es demana els camps necessaris: nom, cognoms, import i data de pagament (per si es vol apuntar una transacció pendent).
3. Omple els camps obligatoris i dona al botó publicar.	4. El sistema compara la contrasenya i l’accepta. Comprova que aquest nom estigui introduït en la base de dades de clients. Si hi es, mostra les dades que s’introduiran i guarda la informació en la taula Banc. informa al administrador de que s’ha actualitzat l’entrada amb èxit.

Post condició: Les dades de usuari es troben disponibles com a expansió de les que ja es troben a les altres taules, sense haver tocat les anteriors, que queden com a historial.

Excepcions:

3. Si l’usuari no ha entrat la contrasenya correcta, el sistema informa de que no es vàlida i retorna al punt 2.

3. Si l’usuari no té assignat el rol d’administrador, no s’introdueix res a la

base de dades, ocasionant un mal funcionament i publicant el resultat sense haver inclòs res a la base de dades.

3. Si l'usuari no existeix en la taula client, no s'introdueixen els valors i es mostra un missatge d'advertència. En aquest missatge s'inclouen els clients amb el mateix primer cognom que l'especificat, per a que al tornar a realitzar la consulta es vegi si s'havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Control d'assistència – Introduir dades:

Un client acaba d'entrar en el recinte i s'ha d'introduir en les llistes d'assistència.

Administrador	Sistema
1. Clica al boto "Control d'assistència: Introduir dades", sota l'apartat Administració del menú.	2. El sistema obre un formulari on es demana els camps necessaris: nom, cognoms i data.
3. Omple els camps obligatoris i dona al botó publicar.	4. El sistema compara la contrasenya i l'accepta. Mostra les dades que s'introduiran i guarda la informació en la taula Asistencia. Informa al administrador de que s'ha actualitzat l'entrada amb èxit. Comprova que l'usuari existeix en la base de dades i si hi és, comprova les seves dades bancàries. Si ha efectuat un pagament en el mes actual, informa al administrador de que ja ha pagat. En cas negatiu, dona un missatge en vermell informant de que no ha pagat encara el mes actual.

Post condició: Les dades de usuari es troben disponibles com a expansió de les que ja es troben a les altres taules, sense haver tocat les anteriors, que

queden com a historial. A més l'administrador pot veure al moment que aquell client encara no ha pagat el mes actual.

Excepcions:

3. Si l'usuari no ha entrat la contrasenya correcta, el sistema informa de que no es vàlida i retorna al punt 2.

3. Si l'usuari no existeix en la taula client, retorna que encara no ha pagat aquest mes. Això és útil ja que de un client que s'ha donat de baixa, pot tornar a assistir reactivant el seu compte al instant.

Consultar estat d'usuari:

Es volen consultar totes les dades associades a un client determinat.

Administrador	Sistema
1. Clica al boto "Consultar Estat d'Usuari", sota l'apartat Consulta del menú.	2. El sistema obre un formulari on es demana els camps necessaris, en aquest cas només el nom.
3. Omple els camps obligatoris i dona al botó publicar.	4. El sistema compara la contrasenya i l'accepta. Comprova que l'usuari existeix en la base de dades i si hi és, descripta les dades sensibles e imprimeix aquesta informació per pantalla. Després comprova les seves dades de taekwondo. En el cas de tenir-ne mostra totes les dades associades de federació i cinturó, de mes nova a mes vella i en cas de tenir l'assegurança caducada, mostra un avis en vermell. Si ha efectuat un pagament en el mes actual, informa al administrador de que ja ha pagat aquest mes. En cas negatiu, dona un missatge en vermell informant de que no ha pagat encara el mes actual.

Post condició: Totes les dades disponibles sobre el client es troben en pantalla. A més l'administrador pot veure al moment que aquell client encara no ha pagat el mes actual i si te l'assegurança de taekwondo en regla, en cas de tenir-ne.

Excepcions:

3. Si l'usuari no ha entrat la contrasenya correcta, el sistema informa de que

no es vàlida i retorna al punt 2.

3. Si l'usuari no té assignat el rol d'administrador, no es mostren els resultats.

3. Si l'usuari no existeix en la taula client, no s'introdueixen els valors i es mostra un missatge d'advertència. En aquest missatge s'inclouen els clients amb el mateix primer cognom que l'especificat, per a que al tornar a realitzar la consulta es vegi si s'havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Consultar dades bancàries d'usuari:

Es volen consultar tot l'historial de pagaments associats a un client determinat.

Administrador	Sistema
1. Clica al boto "Consultar dades bancàries d'Usuari", sota l'apartat Consulta del menú.	2. El sistema obre un formulari on es demana els camps necessaris, en aquest cas només el nom.
3. Omple els camps obligatoris i dona al botó publicar.	4. El sistema compara la contrasenya i l'accepta. Comprova que l'usuari existeix en la base de dades i si hi és, descripta les dades sensibles, ordena aquestes dades en el temps i les mostra en una taula amb l'import i la data de cada mes, en ordre cronològic invers.

Post condició: L'historial de pagaments del client es mostrat en pantalla de manera ordenada.

Excepcions:

3. Si l'usuari no ha entrat la contrasenya correcta, el sistema informa de que

no es vàlida i retorna al punt 2.

3. Si l'usuari no té assignat el rol d'administrador, no es mostren els resultats.

3. Si l'usuari no existeix en la taula client, no s'introdueixen els valors i es mostra un missatge d'advertència. En aquest missatge s'inclouen els clients amb el mateix primer cognom que l'especificat, per a que al tornar a realitzar la consulta es vegi si s'havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Consultar assistències d'avui:

Cada canvi de torn, es posen en comú les assistències que hi ha hagut durant l'estada dels diferents administradors.

Administrador	Sistema
1. Clica al boto "Assistències d'avui", sota l'apartat Consulta del menú.	2. El sistema comprova quin dia és avui i mostra les dades en una taula ordenada en el temps, de més aviat a mes tard.

Post condició: Es mostra per pantalla una taula amb les assistències d'avui especificant l'hora de cadascuna. S'ha fet així per poder accedir a aquesta presentació amb dos clics.

3.1.3.3 Casos d'ús Gamificació:

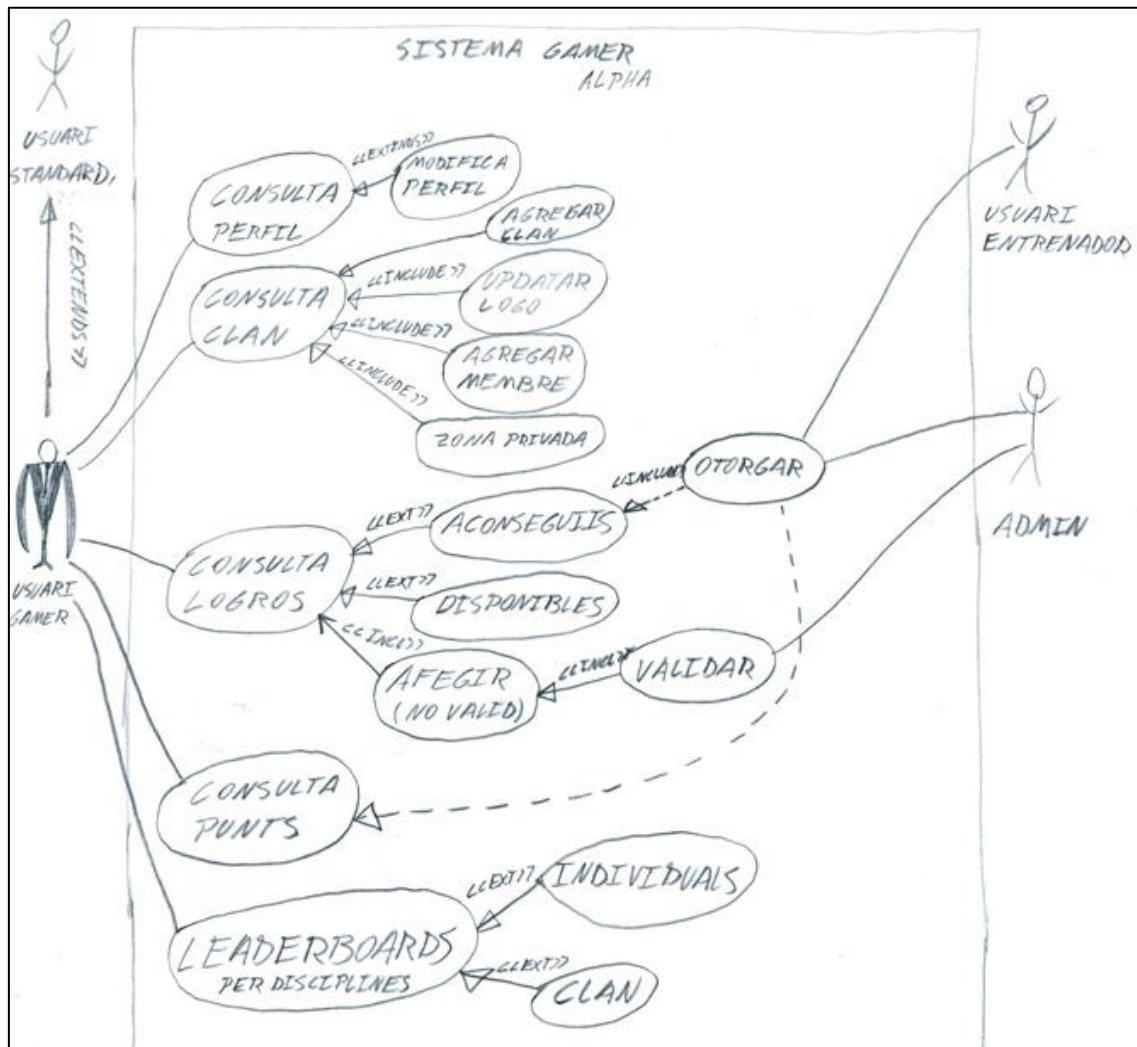


Figura 9: Casos d'us Gamificació

Aquest escenari és una proposta en fase *alfa*, això vol dir que s'anirà expandint i afegint funcionalitats en fases posteriors. En aquest apartat, les tasques es divideixen en tasques d'usuari (com consultar el seu perfil), i tasques d'administrador (com introduir les noves dades), depenent de a quin rol es troben associades. Els apartats d'administració, com en anteriors apartats, es troben en el menú Administrar, mentre que els apartats dels usuaris registrats es troben en el menú JUGA.

La visualització de les dades dels campionats es fa clicant la categoria des del submenú JUGA. S'obre una pagina nova amb els resultats actuals de la classificació: els tres primers en una graella amb medalles d'or, plata i bronze, i la resta en una taula amb les posicions, tots ells amb el rècord que

ostenten actualment i la data en la que ho van aconseguir.

Afegir dades campionat - Abdominals:

Un usuari ha fet uns exercicis i vol que s'incloguin en les taules classificatòries.

Administrador	Sistema
1. Clica al boto “Afegir dades: Abdominals” , sota l’apartat Campionats del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris: nom, cognoms i repeticions de l’exercici.
3. Omple els camps obligatoris i dona al botó publicar.	4. Comprova que aquest nom estigui introduït en la base de dades de clients. Si hi és, mostra les dades que s’introduiran, les guarda en la taula historial del campionat d’abdominals i informa l’administrador de que s’han introduït amb èxit.

Post condició: Les noves dades es troben en la taula de campionat d’abdominals.

Excepcions:

3. Si l’usuari no té assignat el rol d’administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

3. Si l’usuari no existeix en la taula client, no s’introdueixen els valors i es mostra un missatge d’advertència. En aquest missatge s’inclouen els clients amb el mateix primer cognom que l’especificat, per a que al tornar a realitzar la consulta es vegi si s’havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Afegir dades campionat - Peses:

Un usuari ha fet uns exercicis amb pesos i vol que s'incloguin en les taules classificatòries.

Administrador	Sistema
1. Clica al boto “Afegir dades: Peses”, sota l’apartat Campionats del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris, nom, cognoms, pes i repeticions.
3. Omple els camps obligatoris i dona al botó publicar.	4. Comprova que aquest nom estigui introduït en la base de dades de clients. Si hi és, mostra les dades que s'introduiran, les guarda en la taula historial del campionat de pesos i informa l'administrador de que s'han introduït amb èxit.

Post condició: Les noves dades es troben en la taula de campionat de pesos.

Excepcions:

3. Si l'usuari no té assignat el rol d'administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

3. Si l'usuari no existeix en la taula client, no s'introdueixen els valors i es mostra un missatge d'advertència. En aquest missatge s'inclouen els clients amb el mateix primer cognom que l'especificat, per a que al tornar a realitzar la consulta es vegi si s'havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Afegir dades campionat - Bicicleta:

Abans de cap exercici és obligatori escalfar. Molts prefereixen la bicicleta estàtica per a aquest tema, i alguns directament hi competeixen.

Administrador	Sistema
1. Clica al boto “Afegir dades: Bicicleta estàtica” , sota l’apartat Campionats del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris, nom, cognoms i quilòmetres.
3. Omple els camps obligatoris i dona al botó publicar.	4. Comprova que aquest nom estigui introduït en la base de dades de clients. Si hi és, mostra les dades que s’introduiran, les guarda en la taula historial del campionat de bicicleta estàtica i informa l’administrador de que s’han introduït amb èxit.

Post condició: Les noves dades es troben en la taula de campionat de bicicleta estàtica.

Excepcions:

3. Si l’usuari no té assignat el rol d’administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

3. Si l’usuari no existeix en la taula client, no s’introdueixen els valors i es mostra un missatge d’advertència. En aquest missatge s’inclouen els clients amb el mateix primer cognom que l’especificat, per a que al tornar a realitzar la consulta es vegi si s’havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Afegir dades campionat - *Squash*:

L’*Squash* en si mateix ja es un esport de competició, i els usuaris que disputen partits en el centre ja tenen lligues pròpies.

Administrador	Sistema
1. Clica al boto “Afegir dades: Squash” , sota l’apartat Campionats del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris, nom i cognoms dels dos jugadors, puntuació de cadascun i data del partit.
3. Omple els camps obligatoris i dona al botó publicar.	4. Comprova que aquests noms estiguin introduïts en la base de dades de clients. Si hi són, mostra les dades que s’introduiran, les guarda en la taula historial del campionat d’squash i informa l’administrador de que s’han introduït amb èxit.

Post condició: Les noves dades es troben en la taula de campionat de squash.

Excepcions:

3. Si l’usuari no té assignat el rol d’administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

3. Si els l’usuaris no existeixen en la taula clients, no s’introdueixen els valors i es mostra un missatge d’advertència. En aquest missatge s’inclouen els clients amb el mateix primer cognom que els especificats, per a que al tornar a realitzar la consulta es vegi si s’havia equivocat o els clients es troben registrats amb un altre nom (Exemple de dualitat Paco-Francisco).

Afegir fita:

Les fites són elements propis de qualsevol sistema de joc. Influeixen en la motivació dels usuaris per a que continuïn jugant. En fases posteriors es permetrà que els usuaris publiquin les seves pròpies fites, tot i que de moment al ser una qüestió sensible i haver menors implicats, només els

poden crear els administradors.

En l'exemple, es vol afegir una fita anomenada *First Blood* que es donarà als usuaris que per primer cop es caiguin de la cinta de córrer.

Administrador	Sistema
1. Clica al boto “Afegir fita” , sota l’apartat Campionats del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris, nom de la fita, descripció de la mateixa i el nom de la imatge associada a ella (la medalla en qüestió).
3. Omple els camps obligatoris i dona al botó publicar.	4. Comprova que aquest nom no estiguin introduït en la taula de fites. Si no hi és, mostra les dades que s’introduiran, les guarda en la taula historial fites i informa a l’administrador de que s’han introduït amb èxit.

Post condició: Les noves fites poden ser seleccionades en el desplegable del menú “associar fita a usuari” i es troben en la taula de fites.

Excepcions:

3. Si l’usuari no té assignat el rol d’administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

3. Si el nom de la fita ja existeix en la taula de fites, no s’introdueixen els valors i es mostra un missatge d’advertència.

Associar fita a usuari:

Les fites introduïdes en la tasca anterior, serveixen per a ser associades als

usuaris i que aquests les puguin veure i compartir.

Administrador	Sistema
1. Clica al boto “Associar Fita a Usuari” , sota l’apartat Campionats del menú.	2. El sistema obre un formulari de registre on es demana els camps necessaris: nom, cognoms i fita (que és un desplegable de les fites que actualment hi ha a la taula de fites.
3. Omple els camps obligatoris i dona al botó publicar.	4. Comprova que aquest nom estigui introduït en la base de dades de clients. Si hi és, mostra les dades que s’introduiran, les guarda en la taula historial de fites dels usuaris i informa l’administrador de que s’han introduït amb èxit.

Post condició: Les noves dades es troben en la taula de fites d’usuari.

Excepcions:

3. Si l’usuari no té assignat el rol d’administrador, no es fa cap consulta a la base de dades, ocasionant un mal funcionament i retornant les dades, però sense haver-les inclòs a la base de dades.

3. Si l’usuari no existeix en la taula client, no s’introdueixen els valors i es mostra un missatge d’advertència. En aquest missatge s’inclouen els clients amb el mateix primer cognom que l’especificat, per a que al tornar a realitzar la consulta es vegi si s’havia equivocat o el client esta registrat amb un altre nom (Exemple de dualitat Paco-Francisco).

Observar perfil:

Les fites obtingudes i els exercicis que ha realitzat l’usuari poden ser visualitzats des del menú JUGA, apartat “Observa el teu Perfil!”.

Usuari	Sistema
1. Clica al boto “ Observa el teu Perfil! ” , sota l’apartat JUGA del menú principal.	2. El sistema obre la pàgina de l’usuari, mostrant la seva imatge de gamificació (diferent de la del perfil), El seu nom, el seu nivell (depenent de les fites fixes que ha assolit), els dies que ha assistit en total, un llistat de les medalles que ha obtingut (tant temporals com fixes) i el llistat de les fites fixes que li queden per obtenir .

Post condició: L’usuari troba una pàgina amb les seves fites.

Excepcions:

3. Si l’usuari no te inclòs el DNI en el seu perfil de *Drupal* (tot i que no es públic i no es demana al formulari de registre) apareixerà un missatge informant de que ha de demanar als administradors del gimnàs que li activin el apartat de gamificació per al seu perfil.

3. Algunes de les fites són variables o dependents d’altres (com són les fites de nivell, que comproven el nombre de fites assolides sobre el total i puntuen a l’usuari).

3.2 Prototipatge:

Els requeriments definits i la idea general de l’estructura que es vol executar, fan necessària una pausa per realitzar un prototip que avalui la funcionalitat d’aquestes característiques i començar a idear el seu aspecte visual. Són documents, dissenys o sistemes que simulen o tenen implementades parts del sistema final, constituint un per a fer participar l’usuari en el desenvolupament i poder avaluar el producte des de les primeres fases.

Es proposa un prototipatge horitzontal, el qual inclou tota la interfície de

les característiques del sistema, però no conté funcionalitat.

3.2.1 Prototip de paper:

Per començar, es realitza un prototip de baixa fidelitat en paper i llapis (Figura 11 i 12), sense colors, per a donar una idea de l'aspecte general de la web, dissenyar les interfícies i comprovar que l'usuari es capaç d'interactuar amb el disseny proposat. Es realitza una prova d'ús estil *thinking aloud* (pensar en veu alta, figura 10) amb un dels administradors, enregistrada en video (es troba en el directori *1-Prototipatge/paper* del DVD)

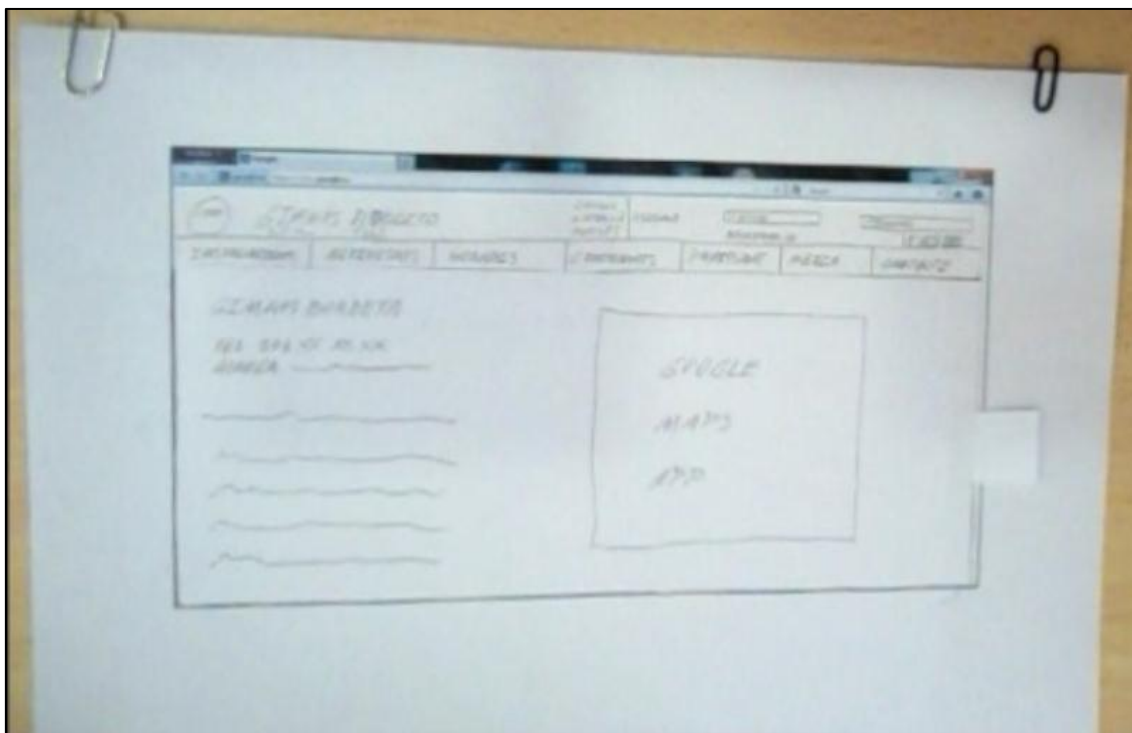


Figura 10: Prototip en Paper

Com a resultat, es proposen una sèrie de canvis en la estructura de menús, ja que resulten massa extensos. Es decideix reagrupar les activitats per a que no hi hagi més de vuit nodes fulla per cada branca. Així les Arts Marcials s'engloben en una categoria pròpia, separades de les classes dirigides. També es separa el formulari de contacte, per a que estigui a la vista des de qualsevol de les pàgines i s'afegeix el apartat Altres Serveis, que no s'havia parlat durant l'anàlisi de serveis que ofereix el centre. A més s'havia incorporat un nou apartat: Taekwondo, per trobar mes de-

pressa la informació al respecte.

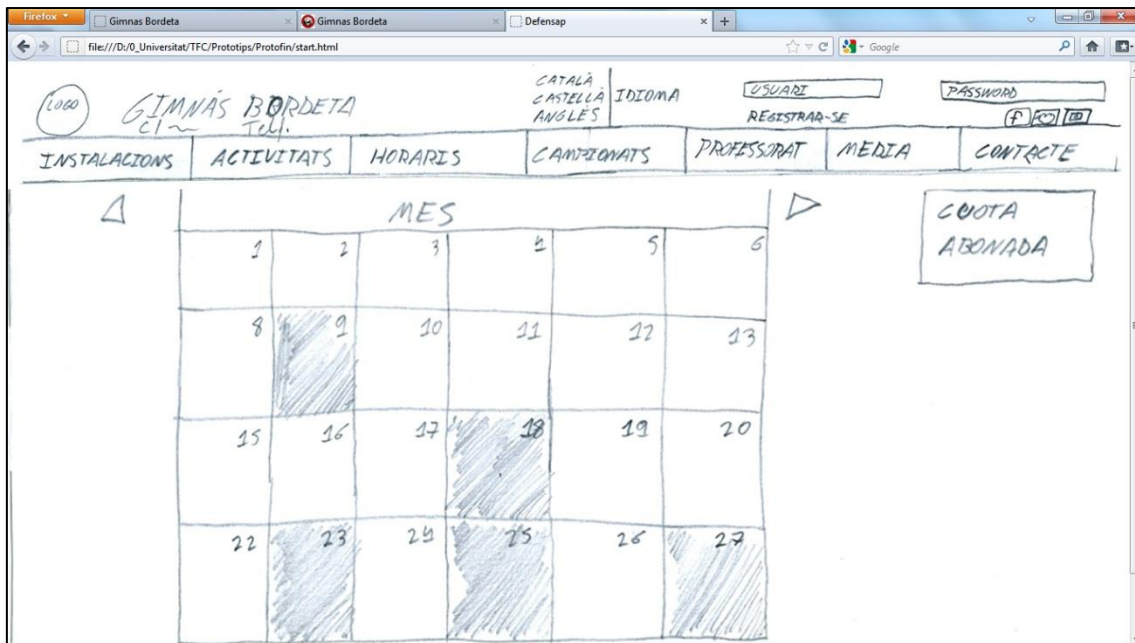


Figura 11: Prototip en paper: Horaris

Alguns usuaris troben angoixant veure en totes les pàgines el missatge “Quota pendent d’abonar”. Davant el risc de que no s’identifiquin per aquest motiu, només ho podran veure i consultar els administradors.

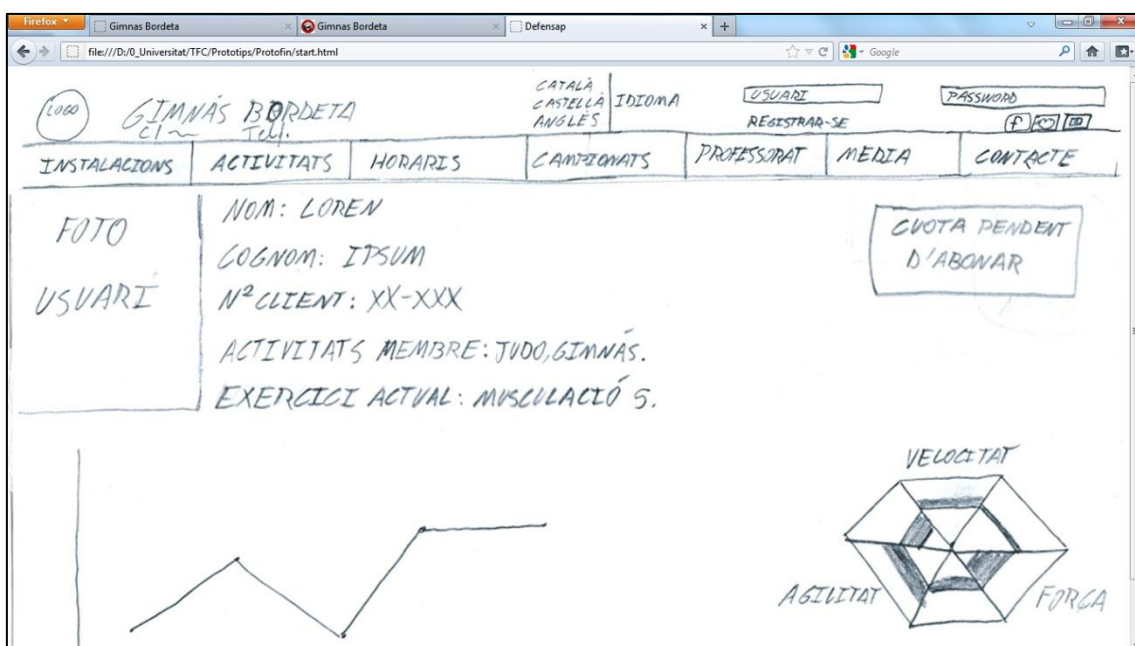


Figura 12: Prototip en Paper: Compta d'usuari

3.2.2 Proves de color:

El següent pas es realitzar diferents proves amb l'esquema de colors definit als requeriments, donant diferent valor als elements i comprovant com s'integren entre ells, es construeixen quatre models diferents mitjançant *photoshop*. Tot i que tots compleixen les normes de contrast vistes prèviament i utilitzen els mateixos colors, són molt diferents. Les dues primeres mostren excessius colors, que carreguen la vista, a més de que donen una idea poc seriosa del centre (figures 13 i 14). Tot i això, comparant amb el *banner* de la figura 15, s'escull el model on predomina el vermell (color corporatiu del centre).

Per últim es proposa un disseny en gris clar del fons dels continguts (figura 16), que resulta més relaxat a la vista i dona més sensació d'espai.



Figura 13: Proves de color (Fred)



Figura 14: Proves de color (Càlid)



Figura 15: Proves de color (Banner gris)



Figura 16: Proves de color (Final)

3.2.3 Prototip Digital:

L'estructura del portal està clara (amb els canvis aplicats en el model de paper) i l'esquema de colors definit. En aquest punt, es fa la primera implementació digital del treball, amb el software de prototipatge *Axure* (www.axure.com) en la seva versió 6.5 (licència de prova d'un mes), que exporta un model funcional (tot i que increïblement redundant i ineficient, figura 17) del portal. Al ser només un model de prova, no inclou molts dels elements que ha de tenir el resultat final, però tot i això és un model d'alta fidelitat que ens ajudarà a modelar alguns aspectes de la implementació.

Consta únicament dels apartats estàtics del portal, per comprovar amb els usuaris l'esquema de colors, l'ordre i la interacció (figura 18).

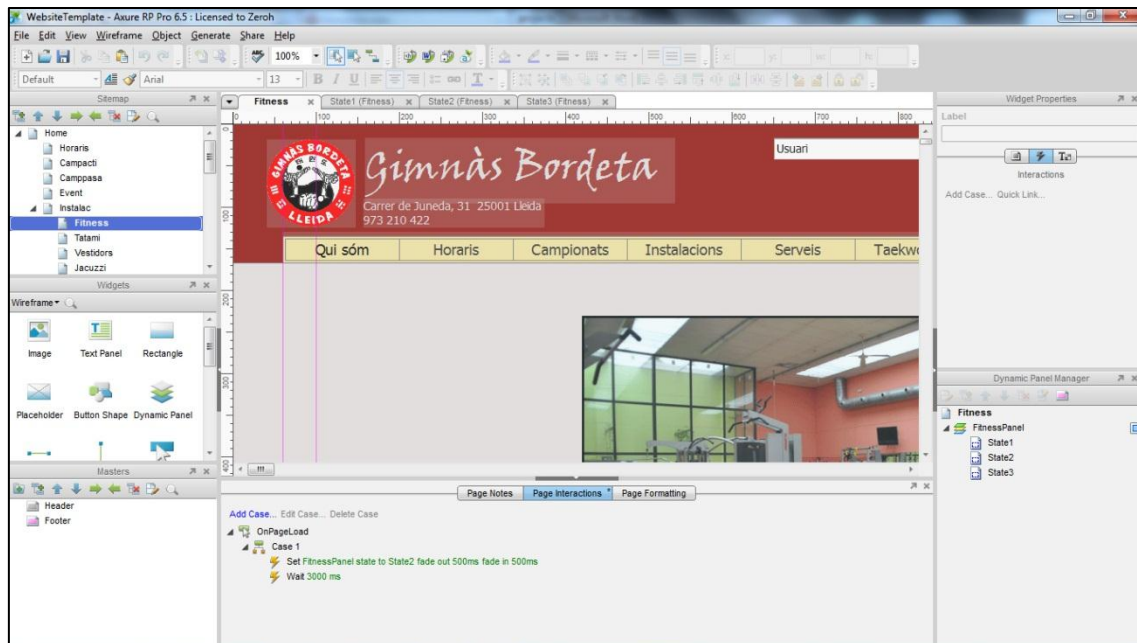


Figura 17: Interfície d'Axure



Figura 18: Export d'Axure en HTML

3.2.3.1 Colors:

L'esquema de colors escollits agrada a tothom (tant usuaris com administradors) ja que incorpora d'una manera eficaç els colors corporatius

del gimnàs que s'integren en l'entorn del centre (utilitza colors similars als de les parets de les instal·lacions). Tot i això, es canvia el color dels elements del menú principal (figura 19) per resultar incòmodes a la vista (el groc resulta un pel incòmode en períodes llargs), decidint-se per un model més clar (blanc amb lletres negres pels desmarcats, gris fosc amb lletres blanques pels marcats). També s'incrementa el contrast entre el fons i el text, definint el color de fons dels continguts a blanc amb lletres negres, un model més clàssic i minimalista.



Figura 19: Banners

També es defineixen després de l'experiència tres àrees de color diferenciades, per a que els usuaris se n'adonin de que han canviat de tipus de tasca. Les tasques de la web principal tindran un *banner* amb un degradat de color vermell, les tasques d'administració un *banner* de color

blau i un indicador *admin* davall del nom del gimnàs i les tasques de gamificació per als usuaris un degradat de gris a blanc, amb l'etiqueta *Gaming* sota el nom del gimnàs. A més al entrar en parts interactives, canvia lleugerament l'alineació dels elements per emfatitzar el canvi de situació.

3.2.3.2 Logotips:

Per tal de poder canviar l'aspecte dels *banners* utilitzant el mateix logotip, s'ha escollit modificar el logotip del gimnàs (que porten cosits en els cinturons i en les samarretes dels entrenadors) tornant tots els colors excepte el negre i el blanc en transparència, per tal d'integrar-lo amb en el fons que hi hagi.



Figura 20: Logotip

Els logotips de les medalles són variacions d'aquest logo, amb el cercle exterior amb una textura d'or, plata o bronze i una imatge interior representativa de la competició actual. Les fites per un altra banda també tenen les seves pròpies medalles, amb el contorn en un color que defineix el tipus de fita (exemple verd positiva, vermell negativa) i una imatge que la representa.



Figura 21: Medalles

3.2.3.3 Tipografia:

L'arxiu d'estils tindrà varies possibilitats de tipus de lletra, en cas de que no estigui disponible en el dispositiu de l'usuari ("*Helvetica Neue*", *Helvetica*, *Arial* i *sans-serif*), tot i que per al títol de la pàgina es descarrega la tipografia *Viner Hand ITC*, la qual es troba en el servidor per si l'usuari no disposa d'ella.

3.2.3.4 Organització:

Pel que fa a l'organització, els usuaris troben que l'apartat taekwondo hauria d'estar englobat en l'apartat Arts Marcials, on el buscarien d'entrada, així que es torna a canviar (tot i que s'havia pensat en la possibilitat de posar dos links per a una mateixa secció, es descarta per trobar-se una solució excessivament confusa).

També es defineixen el tipus d'*slideshow* que es vol pels apartats que en continguin un (a la maqueta n'hi ha dos de diferents, en la figura 18 es veu un dels tipus descartats), en concret el que te *thumbnails* amb una imatge cíclica que canvia cada pocs segons, excepte si l'apuntes amb el ratolí.

A més serveix per a que els administradors del gimnàs vegin quina serà la informació que hauran de subministrar per tal de emplenar els diferents apartats amb text i imatges.

El prototip sencer es pot trobar en el directori *1-Prototipatge/Digital/Protofin/index.html*.

3.3 Previ a implementar:

El portal tindrà parts interactives per als usuaris i administratives per als propietaris, per tant requerirà d'una base de dades on guardar informació (i des d'on consultar-la). Tot i que la arquitectura específica s'establirà en la implementació, es fa una reunió amb els gestors del gimnàs per esbrinar exactament amb quines dades dels usuaris volen interactuar.

En una primera fase, la funcionalitat que es busca des de l'administració és:

Una taula principal, on guardar les dades dels clients, separada de la taula d'usuaris registrats a la web. El motiu es que no volen que un tingui que estar registrat obligatòriament a la web per poder visualitzar les dades i gestionar-lo. Contindrà el número de matriculació que dona el gimnàs, el seu NIF (o NIE), el nom complet del client, la ciutat on resideix amb la seva adreça corresponent, la seva data de naixement (per calcular la edat en cas de ser necessari), data de matriculació, el mode de pagament que utilitzarà (domiciliació bancària o efectiu) i un possible comentari curt sobre el client.

L'apartat Banc només serà funcional en cas de que el mode de pagament sigui domiciliació bancària. Llavors es relacionarà el NIF o NIE amb la quantitat pagada i el número de compte corrent corresponent.

Una de les activitats més importants del gimnàs són les competicions de taekwondo, disputades sota la gestió de la Federació Catalana de Taekwondo. Per participar en les activitats del gimnàs al respecte (fins i tot per assistir a les classes) s'ha d'estar federat. Els motius tenen a veure amb que la federació dona un segur a tot risc que més d'un cop ha ajudat a algú amb alguna lesió, des les més lleus a les greus. Per tant, alguns dels seus clients tindran dades de federació (que s'inclouran en la taula Taekwondo). Aquestes dades són el número de federació, que te data de caducitat i s'ha d'anar renovant, i el cinturó actual, associat a la data amb la que s'ha aconseguit.

Aquestes dades han de ser de fàcil consulta, ja que de vegades es necessària alguna d'elles en concret (emplenat els papers de l'asseguradora en cas de

lesió o inscriure als membres del gimnàs en un campionat).

A més, es porta un registre dels assistents diaris al centre (únicament l'hora d'entrada, mai la de sortida). Tenint en compte que una de les modalitats de pagament és en efectiu, el centre ha expressat el seu interès en que a l'hora d'informatitzar el control d'assistències, es visualitzi si el client ha efectuat ja el pagament durant el mes actual. Tot això anirà a la taula assistència.

Per gestionar els resultats de les competicions internes actives, s'utilitzaran taules individuals. Les noves fites, premis en forma de medalla al assolir resultats, poden ser introduïdes per un administrador i, per als clients que ho vulguin, hi ha la opció d'associar-les a un usuari. Aquest serà l'apartat de gamificació.

En una segona fase, com a treballs futurs, es continuarà amb la implementació de noves tasques, com ara la digitalització de les taules d'exercicis de *fitness* i la possibilitat de que cada usuari creï les seves. A més es continuarà ampliant els elements de joc per als usuaris, amb la creació de clans i l'augment en les possibilitats dels usuaris, com crear les seves pròpies fites i medalles i donar-les als usuaris que ells vulguin.

Bloc 4:

Desenvolupament

En el bloc 2 s'han definit les necessitats que s'han de satisfer i en el bloc 3 s'han concretat les diferents tasques i el disseny que ha de tenir tant el portal com els diferents apartats amb els que els usuaris poden interactuar.

Aquest projecte s'implementa sobre *Drupal 7*, que de per si mateix ja té algunes de les funcionalitats exigides. Per aquest motiu aquest bloc no serà només fase de codificació, doncs part del codi ja ha estat escrit per altres desenvolupadors aliens a aquest projecte. Degut al seu disseny modular, i a que cada mòdul crea taules pròpies en la base de dades s'escull treballar en dos sistemes alhora: local i online. En el sistema local es provaran les funcionalitats dels mòduls i s'escollirà el millor, que s'implementarà en el sistema online un cop es vegi que s'obté el resultat desitjat. Això permet a més tenir un portal publicat i funcional sense brossa o parts en desenvolupament. A més la copia local es exacta a l'estat de la copia online (excepte pels continguts d'algunes taules de la base de dades, de les quals es fan còpies cada setmana i s'inclouen en la local).

El servei online es contracta a *Nominalia*, en concret el “*hosting premium*”, que ofereix entre d'altres coses per prop de 190 euros/any: 1.000 GB de tràfic mensual, sense límit d'ample de banda; suport per a *php* 4.4.9 - 5.3.3; 50 bases de dades *MySQL* amb 1 GB d'espai per aquestes; 500 comptes de correu pel domini.

Utilitzant aquest sistema redundant es pràcticament impossible perdre la totalitat de les dades contingues en el sistema.

4.1 Disseny del portal:

Un cop instal·lat *Drupal*, es procedeix a dissenyar primer el tema del portal en si mateix, per a que tingui l'aspecte gràfic prototipat en el bloc anterior. Un cop assolit, es van incloent i retocant mòduls per disposar de totes les funcionalitats llistades en els casos d'us.

4.1.1 Tema:

El disseny del portal es duu a terme mitjançant la modificació del tema de *Drupal 7* anomenat “*Bartik*” (figura 22), el qual té un aspecte per defecte similar al que es pretén obtenir.

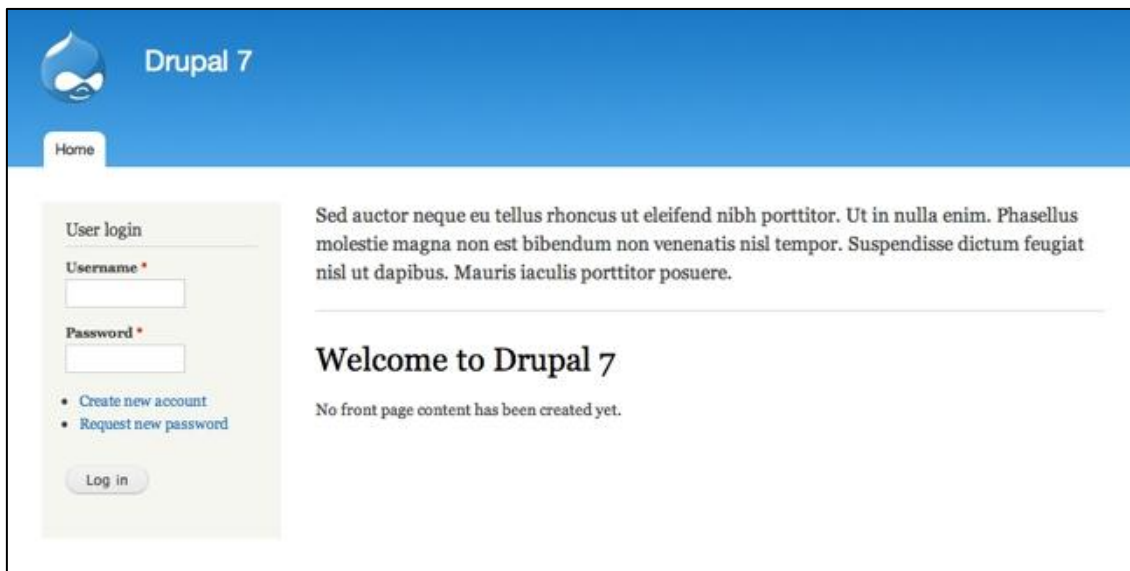


Figura 22: *Bartik*

4.1.1.1 Esquema de colors:

Es procedeix primerament a canviar l'esquema de colors. Això es realitza fàcilment sense tocar codi a través de l'assistent de *Drupal* situat a Administració » Aparença » Paràmetres:

S'utilitzen els colors seleccionats en els requisits previs. Aquest assistent modifica l'arxiu d'estils *colors.css* i en guarda una còpia a part per a posteriors recuperacions. Tot i això, alguns elements es tenen que modificar directament en l'arxiu *style.css*, on es contenen tots els estils de tipografia, taules, imatge, botons i disseny. El valor del color de les taules es treu per a que



Figura 23: Esquema de colors

respecti el color de fons del contingut i es defineix un ample del 100% per a que ocupi la totalitat de l'ample del paràgraf, adaptant-se a la resolució de l'usuari.

Amb l'assistent tampoc es pot diferenciar l'esquema de colors segons l'apartat on ens trobem, així que la modificació prevista pels apartats d'administració i gamificació no podria dur-se a terme. Per solventar això s'afegeix una modificació a l'arxiu *colors.css*, afeguem dos elements *aheader* i *gheader* amb els colors projectats.

```
#gheader {
    background-color: #000000;
    background-image: -moz-linear-gradient(top, #000000 10%, #d6d6d6 90%);
    background-image: -ms-linear-gradient(top, #000000 10%, #d6d6d6 90%);
    background-image: -o-linear-gradient(top, #000000 10%, #d6d6d6 90%);
    background-image: -webkit-gradient(linear, left top, left bottom, color-stop(0, #000000), color-stop(1, #d6d6d6));
    background-image: -webkit-linear-gradient(top, #000000 10%, #d6d6d6 90%);
    background-image: linear-gradient(top, #000000 10%, #d6d6d6 90%);
}

#aheader {
    background-color: #000000;
    background-image: -moz-linear-gradient(top, #000000 10%, #16268b 90%);
    background-image: -ms-linear-gradient(top, #000000 10%, #16268b 90%);
    background-image: -o-linear-gradient(top, #000000 10%, #16268b 90%);
    background-image: -webkit-gradient(linear, left top, left bottom, color-stop(0, #000000), color-stop(1, #16268b));
    background-image: -webkit-linear-gradient(top, #000000 10%, #16268b 90%);
    background-image: linear-gradient(top, #000000 10%, #16268b 90%);
}
```

La constructora de la pàgina no contempla la possibilitat d'introduir aquestes noves dades, ja que a l'arxiu *page.tpl.php*, encarregat de la codificació de l'estructura de la pàgina que s'envia a l'usuari, s'especifica que tota pàgina comença amb una etiqueta de classe "header":

```
# <div id="page-wrapper"><div id="page"><div id="header"
```

Per a diferenciar la constructora segons el tipus de contingut especificat,

s'ha d'editar l'arxiu *template.php*, que conté una funció per afegir precondicions en la generació del codi *html*, on s'insereix:

```
function bartik_process_page(&$variables) {
  if (isset($variables['node']->type)) {
    $nodetype = $variables['node']->type;
    $variables['theme_hook_suggestions'][] = 'page__' . $nodetype;}
  [...]
}
```

Aquest codi fa que abans d'aplicar la constructora, comprovi dins del directori de constructores del *template* que no hi hagi un arxiu *page--[tipus de contingut].tpl.php* i en cas de ser-hi, l'aplica en comptes de la constructora per defecte. Això permetrà la creació d'un contingut *admin* i un contingut *gaming*, que carreguin etiquetes d'estil diferents.

4.1.1.2 Tipologia:

L'esquema de colors ja és l'adequat per al portal, però els textos no s'assemblen als prototipats. La marca de l'arxiu d'estils *Site Name* no utilitza el tipus de lletra que volem. El tipus de lletra *Viner Hand ITC* a més és una tipologia que es poc habitual de trobar en els equipaments de la gent, per tant la majoria de navegadors la mostrarien com a una *arial* simple.

Això es solventa posant a l'inici de l'arxiu *css* una ordre per a que es descarregui la tipologia, que prèviament s'ha pujat al servidor.

```
@font-face {
  font-family: 'Customviner'; /*nom de la font pròpia*/
  src: url('http://gimnasbordeta.virgili.org/font/viner.ttf'); /*URL*/
}
[...]
body,
#site-name,
.ui-widget,
.comment-form label {
  font-family: "Customviner", arial;
}
```


4.1.1.3 Logotips i botons:

Amb els colors del portal i els seus elements, i els tipus de lletra establerts, es procedeix a canviar les icones i logotips.

Sota l'apartat Administració » Aparència » Paràmetres, es defineix un logotip diferent al que porta per defecte, en concret el de la figura 17, amb transparències en l'arc exterior per integrar-se amb el fins del *banner*, independentment del seu color. També es crea una icona de 10x10 (🔴) píxels per a mostrar en el navegador el el logotip del centre i s'assigna com a imatge per defecte (La icona d'accés ràpid, o *favicon*, es mostra a la barra d'adreces de la majoria de navegadors).

Per a un ús posterior, s'estableix l'aspecte dels botons que s'utilitzaran per tal de fer els links més atractius visualment. N'hi ha tres de diferents:

- “enlaceboton” serveix per als botons de contacta i altres links que hi pugui haver en els diferents apartats del portal. Pensats per situar-se sobre fons clar, son vermells en repòs i grisos en seleccionat. Tenen uns bordes uniformement rondejats.

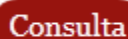


Figura 24: Enlaceboton

```
.enlaceboton {
    padding-top:5px;
    padding-bottom:5px;
    padding-right:10px;
    padding-left:10px;
    background-color: #97140e;
    color: #FFFFFFF;
    text-decoration: none;
}

.enlaceboton:link,
.enlaceboton:visited {

    -khtml-border-radius-topleft: 15px;
    -khtml-border-radius-topright: 15px;
    -moz-border-radius-topleft: 15px;
```



```

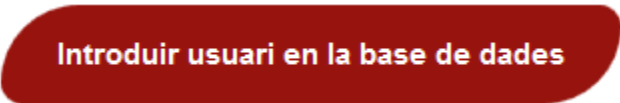
-moz-border-radius-topright: 15px;
-webkit-border-top-left-radius: 15px;
-webkit-border-top-right-radius: 15px;
border-top-left-radius: 15px;
border-top-right-radius: 15px;

-khtml-border-radius-bottomleft: 15px;
-khtml-border-radius-bottomright: 15px;
-moz-border-radius-bottomleft: 15px;
-moz-border-radius-bottomright: 15px;
-webkit-border-bottom-left-radius: 15px;
-webkit-border-bottom-right-radius: 15px;
border-bottom-left-radius: 15px;
border-bottom-right-radius: 15px;

}
.enlaceboton:hover {
    background-color: #9d9c9c;
    color: #000000;
}

```

· “boto” serveix per als botons del mòdul d’administració i per a seleccionar els campionats de gamificació. Tenen les mateixes característiques que “enlacebotón”, però en canvi els seus bordes són rodrejats amb simetria diagonal.



Introduir usuari en la base de dades

Figura 25: Boto

```

.boto:link,
.boto:visited {
[... ]
-khtml-border-radius-topleft: 55px;
-khtml-border-radius-topright: 15px;
-moz-border-radius-topleft: 55px;
-moz-border-radius-topright: 15px;
-webkit-border-top-left-radius: 55px;
-webkit-border-top-right-radius: 15px;

```

```
border-top-left-radius: 55px;
border-top-right-radius: 15px;

-khtml-border-radius-bottomleft: 15px;
-khtml-border-radius-bottomright: 55px;
-moz-border-radius-bottomleft: 15px;
-moz-border-radius-bottomright: 55px;
-webkit-border-bottom-left-radius: 15px;
-webkit-border-bottom-right-radius: 55px;
border-bottom-left-radius: 15px;
border-bottom-right-radius: 55px;

}
```

· “botobanner” es va introduir en una fase posterior davant la necessitat de posar la funcionalitat “Contacta” en un punt concret en totes les possibles vistes del portal. És idèntic a “enlacebotón”, però te els colors per a seleccionat des-seleccionat invertits respecte el mateix, per tal d’integrar-se amb el fons del banner.

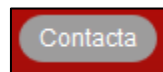


Figura 26: Botobanner

```
.botobanner {
    padding-top:5px;
    padding-bottom:5px;
    padding-right:10px;
    padding-left:10px;
    background-color: #9d9c9c;
    color: #000000;
    text-decoration: none;
}

[...]

.botobanner:hover {
    background-color: #ffffff;
    color: #97140e;}
```

En els diferents nodes que es van creant, apareixerà l’anomenat *breadcrumb*, que és una llista de links que ens mostra el camí de pàgines

que s'ha seguit per arribar al node actual, des del node Inici (la pàgina de portada). Tot i poder arribar a ser útil, com l'estructura de menús és realment clara i senzilla, es decideix prescindir d'aquesta funció. Per tal de inhabilitar-la es té que modificar la constructora de nodes, en concret la que s'ocupa de les capçaleres i menús *page.tpl.php*, eliminant el codi:

```
[...]
if ($breadcrumb == true){
print $breadcrumb;
}
[...]
```

D'aquesta manera no reservarà espai en blanc entre el menú principal i els continguts.

Drupal ofereix un mòdul de gestió de menús integrat, però el *template Bartik* no permet la utilització de branques en l'estructura de menús, i per tant no els mostra. Tot i que hi ha multitud de mòduls que es poden afegir per obtenir aquesta funcionalitat, es decideix extreure el codi necessari per a fer funcionar els actuals. El mòdul de menús més representatiu és *nice menús* que utilitza els mòduls de *jQuery superfish*. Es comprova que *superfish* es pot adaptar al tema seguint els passos descrits a continuació:

S'afegeix a l'arxiu de preprocès *template.php* una funció per a que retorni l'arbre de menús:

```
/**
 * Funció per desplegar menus.
 */
function bartik_links($variables) {
    if(array_key_exists('id', $variables['attributes']) &&
    $variables['attributes']['id'] == 'main-menu-links') {
        $pid = variable_get('menu_main_links_source', 'main-menu');
        $tree = menu_tree($pid);
        return drupal_render($tree);
    }
    return theme_links($variables);
}
```

Es descarrega de la web de *jQuery* l'script *superfish.js* i es guarda en un

directori dins del *template* juntament amb un arxiu (s'anomenarà *bartikmenu.js*) que conté les qualitats dels menús que es generaran, amb el següent codi:

```
// $Id
Drupal.behaviors.bartikBehavior = {
  attach: function(context) {
    /**
     * http://users.tpg.com.au/j_birch/plugins/superfish/
     * @see js/superfish.js
     */
    jQuery('.navigation ul').superfish({
      animation: { opacity: 'show', height:'show' },
      easing: 'swing',
      speed: 250,
      autoArrows: false,
      dropShadows: false /* Needed for IE */
    });
  }
};
```

Ara únicament s'ha de modificar els valors per defecte dels *css* que gestionen els menús per a que incloguin els elements desplecats:

```
#main-menu .menu ul {
  background: #fff;
  border: solid 1px #ddd;
  line-height: 1.25em;
  color: #555;
  display: none; /* Fer que els elements del submenú no es vegin al carregar
la pàgina */
  padding: 0;
  position: absolute;
  top: 36px;
  left: 0;
  z-index: 1000;
}
```

```
#main-menu .menu ul li {
    display: block;
    margin: 0;
    padding: 0;
    position: relative;
    line-height: normal;
    border-right: none;
    text-align: left;
}

#main-menu .menu li .expanded ul {
    margin: -2.65em 0 0 15em;
}

/* Second level links */
#main-menu .menu ul li a,
#main-menu .menu ul li a:visited,
#main-menu .menu li.active-trail ul li a,
#main-menu .menu .sfHover ul li a {
    background: #EEF;
    border-bottom: solid 1px #eee;
    color: #555;
    display: block;
    font-size: 0.9em;
    font-weight: normal;
    overflow: hidden;
    width: 15em;
    line-height: 1.25em;
    padding: 0.5em;
    text-decoration: none;
    -khtml-border-radius-topleft: 0;
    -khtml-border-radius-topright: 0;
    -moz-border-radius-topleft: 0;
    -moz-border-radius-topright: 0;
    -webkit-border-top-left-radius: 0;
    -webkit-border-top-right-radius: 0;
```

```
border-top-left-radius: 0;
border-top-right-radius: 0;
}
/* Borra el borde de dalt */
#main-menu .menu ul li.last a {
    border-bottom: none;
}
/* Mostra una fletxa en cas de que l'element no sigui node fulla */
#main-menu .menu ul li.expanded {
    background: transparent url('images/icn-arrow-right-light.png') no-repeat
    96.5% 50%;
}
/* Borra la fletxa dels nodes fill */
#main-menu .menu li.expanded li.leaf {
    background: none;
}
/* Remarcat en passar per sobre */
#main-menu .menu ul li a:hover,
#main-menu .menu ul li a:visited:hover {
    background: #B3B3B3 url('images/bg-shade-medium.png') repeat-x 0
    100%!important;
    color: #fff!important;
}
/* active submenu link */
#main-menu .menu li.active-trail ul li a.active {
    color: #000;
}
#main-menu .menu ul ul {
    z-index: 1001;
    margin: 0;
    top: 2.65em;
}
```

Únicament resta introduir les crides `scripts[] = scripts/superfish.js` i `scripts[] = scripts/bartikmenu.js` a l'arxiu *bartik.info* per a que carregui els scripts durant l'inici del portal.

S'ha de tenir en compte que per a que es mostri els nodes fulla, en el menú Estructura » Menús » Menú Principal, cal editar tots els nodes pare i clicar en la casella “*Show as expanded*” per a que mostri els nodes fill d'aquell node.

D'aquesta manera *Bartik* ja compta amb menús desplegable, sense haver d'afegir cap mòdul extern.

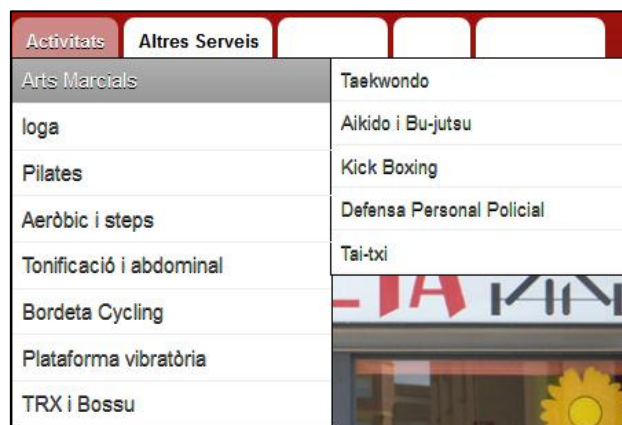


Figura 27: Menús desplegables

4.1.1.4 Ordenació dels blocs:

Amb l'aspecte de colors i tipologia definit, es continua amb l'ordenació dels elements de les pàgines, que no és la prototipada.

Bartik té alguns blocs que no es volen en aquest projecte, o que es volen en un altre lloc. Mitjançant l'administrador de blocs (que es troba a Inici » Administració » Estructura » Blocs) es gestiona la seva posició dins la pàgina.

- Utilitza blocs laterals per mostrar els últims continguts i el menú d'usuari, que no s'utilitza en l'actual projecte. Aquests blocs laterals es treuen, es baixen davall de l'apartat inhabilitat.

BLOC	REGIÓ	OPERACIONS
<div>Últims elements de la categoria Gimnas</div> <div>Bordeta</div>	- Cap -	configura
<div>User menu</div>	- Cap -	configura
Inhabilitat		

Figura 28: Blocs a Inhabilitar

- El *login* d'usuari es troba en un dels menús laterals. Es situa a dalt a la dreta arrastrant l'element “*Inici de sessió d'usuari*” davall de Capçalera.

Nom d'usuari * Contrasenya *

[Crea un compte nou](#) [Demana una contrasenya nova](#)

Figura 29: Login d'usuari

- El contingut del peu no és l'adequat, a l'element “*Funciona amb el Drupal*” se li afegeix el text “*Virgili.org 2013*”, com a marca de qui ha desenvolupat aquest portal.

· *Drupal* no inclou per defecte cap mòdul de gestió de xarxes socials (tot i que si n'inclou un de publicació de *RSS*), i el més recomanat per part dels usuaris per Internet és el mòdul oficial de l'aplicació web *Addthis.com*. Aquest permet incloure els botons de compartir de més de dos-centes xarxes socials diferents, a més de incloure un servei d'estadístiques dels usuaris que hagin utilitzat aquests botons. Lamentablement, un cop implementat en el sistema local s'ha comprovat que no funciona bé l'apartat de compartir en llenguatge català ni castellà, quedant retallades les icones de *Facebook* “*me gusta*”. En aquest punt, es decideix crear a mà un bloc on introduir un script de la web *Addthis.com* i maquetar-lo d'una manera adequada.

Primer de tot es crea un compte a la web, i es va a l'apartat “*get share buttons*”, on surt un assistent que donarà el codi i el *javascript* que es necessitarà per fer els anàlisis dels usuaris.



Figura 30: Asistent Addthis.com

L'important es l'script que dona, amb un codi que va associat a la nostra compta i que permetrà la visualització dels resultats a través de la seva web. En quant als botons, els hem de trobar un a un en l'apartat “*Get the newest buttons*” i integrar-los dins el codi. La proposta van ser les dues xarxes socials més utilitzades pels clients del gimnàs (*Facebook* i *Twitter*), i la xarxa social de *Google* (*google+*) ja que podria adquirir importància amb el temps. A més es posa també un link a *pinterest*, xarxa social de *repost* d'imatges i un desplegable amb totes les xarxes social que algú podria buscar:

```
<div class="addthis_toolbox addthis_default_style ">

<a class="addthis_button_facebook_like" fb:like:layout="button_count"
fb:like:locale="en_US"></a>

<a class="addthis_button_tweet"></a>

<a class="addthis_button_google_plusone" g:plusone:size="medium"></a>

<a class="addthis_button_pinterest_pinit"></a><a class="addthis_counter
addthis_pill_style"></a></div>

<script type="text/javascript">var addthis_config =
{"data_track_addressbar":true};</script>

<script type="text/javascript"
src="//s7.addthis.com/js/300/addthis_widget.js#pubid=ra-
5182b6074de5b587"></script>
```

El resultat te millor separació que el mòdul de *Drupal* perquè també utilitza els *css* del sistema. Es crea un nou bloc nou situat damunt del bloc

“Funciona amb el Drupal” baix l’etiqueta “Peu” i el resultat es aquest:



Figura 31: Compartir Addthis.com

S’aprofita que *Addthis.com* també ofereix botons per a que els clients segueixin les publicacions del centre a les xarxes socials, mantenint l’aspecte visual que s’ha donat en els botons de compartir. El procés és el mateix, tot i que aquest cop s’ha d’especificar el *link* on ha de portat cada botó. S’elegeixen els botons de *Facebook* (única xarxa social on el centre té compta) i de sindicació *rss* del portal (servei que *Drupal* si que ofereix de manera nativa):

```
<div class="addthis_toolbox addthis_32x32_style addthis_default_style">

<a class="addthis_button_facebook_follow"
addthis:userid="gimnas.bordeta"></a>

<a class="addthis_button_rss_follow" addthis:userid="?q=rss.xml"></a></p>

</div>

<script type="text/javascript"
src="//s7.addthis.com/js/300/addthis_widget.js#pubid=ra-
5182b6074de5b587"></script>
```

El mòdul de *Drupal*, curiosament no dona aquesta funcionalitat encara. Es crea un nou bloc anomenat *addthis.com* situat davall del bloc “Inici de sessió d’usuari” baix l’etiqueta “Capçalera” i el resultat es aquest:

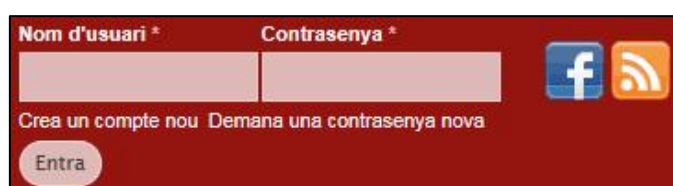


Figura 32: Logos seguir publicacions

4.1.1.5 Disseny Responsiu:

Tot i que la majoria de temes de *Drupal7* estan preparats per a adaptar-se automàticament a múltiples resolucions, el *template Bartik* no contempla aquesta funcionalitat. En l’estat actual, el portal web esta preparat per funcionar únicament utilitzant una resolució mínima de 960

píxels d'ample, per a tota resolució inferior apareix una barra de desplaçament. Mitjançant l'eina web (<http://mattkersley.com/responsive/>) s'emula la presentació en diferents dispositius.

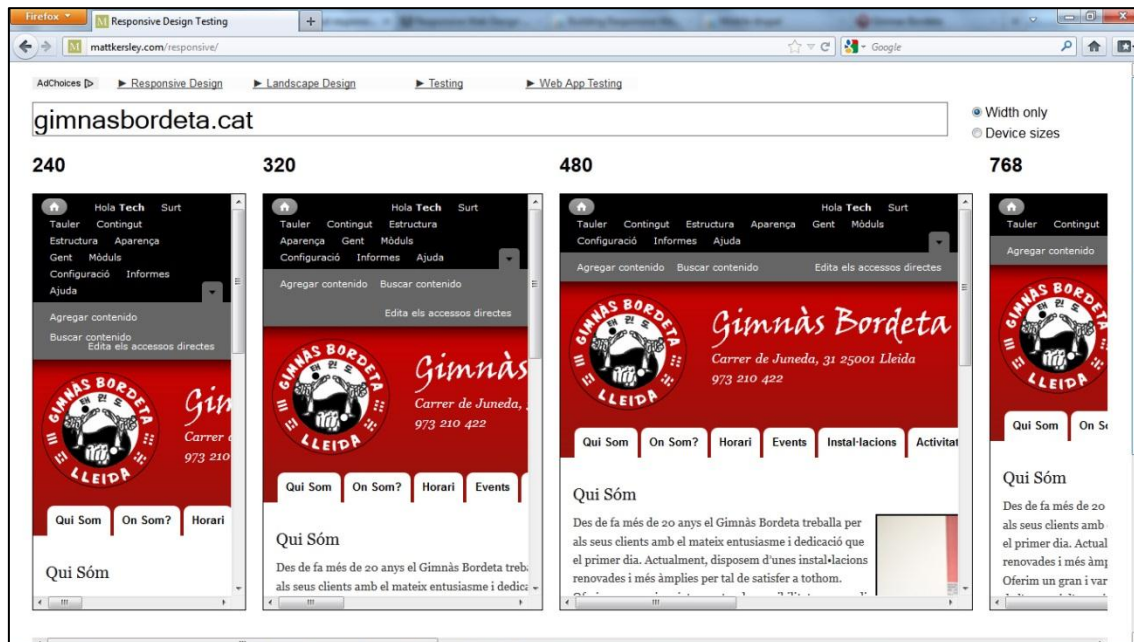


Figura 33: No responsiu

Per tal de millorar la presentació en dispositius mòbils (així com en pantalles més antigues i resolucions per defecte de *tablets* de gamma baixa), s'incorpora a l'arxiu que gestiona quan s'utilitza cada CSS (*bartik.info*) la crida a arxius personalitzats per a resolucions més petites de 320, de 480 i 768, respectivament, a més d'un per a resolucions de 1080p (Alta Definició):

[...]

```
stylesheets[all and (max-width: 320px)][] = css/layout320.css
```

```
stylesheets[all and (min-width: 321px) and (max-width: 480px)][] =  
css/layout480.css
```

```
stylesheets[all and (min-width: 481px) and (max-width: 960px)][] =  
css/layout720.css
```

```
stylesheets[all and (min-width: 961px) and (max-width: 1280px)][] =  
css/layout.css
```

```
stylesheets[all and (min-width: 1281px)][] = css/layoutfhd.css
```

[...]

Aquestes instruccions de *css3* són acceptades per la majoria de navegadors actuals, utilitzant cada fitxer en funció de la resolució de l'usuari en temps real.

En els fitxers per a resolucions inferiors a 960p, s'han substituït totes les mides fixes per percentuals (*width: 960px;* per *width: 100%;*) s'han eliminat les marques de mida mínima (*min-width: 960px;*) per a permetre que els elements es facin més petits en cas de ser necessari. També s'ha reduït els marges entre elements (*margin-top: 5px;*) i l'alineació dels elements s'ha passat a la dreta per a que quedin en columna en les versions de mòbil.

Pel que fa al menú principal, els botons en forma d'etiqueta es descol·loquen en ser reescalats, així que s'ha de modificar l'aspecte dels botons del menú per a les diferents resolucions.

Per als menús de l'estil tablet, es poden reordenar i canviar la forma dels botons utilitzant únicament estils *css*.

En el fitxer *style.css*, s'afegeix el modificador *@media all and (max-width: [x]px)* per a que els usi en funció de la resolució. A més es desactiven els menús flotants, ja que en resolucions petites resulten molestes i és difícil navegar de manera tàctil (s'ofereix un altra solució per a resolucions més petites, una pàgina-menú amb els grans botons que s'han explicat en l'apartat anterior). Els botons resultants seran grans i arrodonits, de fàcil interacció.

```
@media all and (max-width: 460px) {  
    /* ----- Menu Resolució petita ----- */  
  
    /* Aquesta línia es per a que no entri en conflicte amb la solució per a  
    pantalla de telèfon mòbil, descrit a l'apartat següent.*/  
    a#menu-toggle {  
        display: block;  
    }  
  
    #main-menu .menu {  
        display: none;  
        height: auto;  
        text-align: center;  
    }  
}
```

```
#main-menu .menu li {
    float: left;
    margin-right: 5px;
    padding: 0;
    display: inline-block;
    width: 100%;
}

#main-menu .menu li:nth-child(3n) {
    margin-right: -5px; /* LTR */
}

#main-menu .menu a {
    float: none;
    display: block;
    border-radius: 8px;
    margin-bottom: 5px;
    padding: 0.9em 5px;
}

/* ----- DESHABILITAR SUBMENUS-----*/

#main-menu .menu ul {
    display: none;
}

#main-menu .menu ul li {
    display: none;
}

}

@media all and (min-width: 461px) and (max-width: 900px) {
    /* ----- Header and Menus ----- */

    /* Aquesta linea es per a que no entri en conflicte amb la solució per a
    pantalla de telèfon mòbil, descrit a l'apartat següent.*/

    a#menu-toggle {
        display: block;
    }

    #main-menu .menu {
        margin: 0 5px;
    }
}
```

```
padding: 0;
text-align: center;
}
#main-menu .menu li {
float: left;
margin-right: 5px;
padding: 0;
display: inline-block;
width: 32.75%;
}
#main-menu .menu li:nth-child(3n) {
margin-right: -5px; /* LTR */
}
#main-menu .menu a {
float: none;
display: block;
border-radius: 8px;
margin-bottom: 5px;
padding: 0.9em 5px;
}
/* ----- DESHABILITAR SUBMENUS-----*/
#main-menu .menu ul {
display: none;
}
#main-menu .menu ul li {
display: none;
}
}
```

Per a l'aspecte des de les resolucions més petites, es vol ocultar el menú principal i carregar-ne un de desplegable, que ocupi molt poc espai de pantalla mentre no s'utilitzi. La tasca es realitza adaptant un dels tres *scripts* de *jQuery* fets per Lewis Nyman (estudiant que va posar a disposició de tots els usuaris dits *scripts* en la pàgina de *Drupal*), que

serveixen per canviar en temps real etiquetes de la pàgina *html*. Això permet l'eliminació o la reordenació d'elements en el portal sense tenir que tornar a fer cap petició al servidor.

En el fitxer *bartik.info* s'inicialitza l'script per a que l'inclogui al construir la pàgina:

[...]

```
scripts[] = js/collapsible-menu.js
```

[...]

L'script *collapsible-menu.js* es modifica per adaptar-lo a les etiquetes i escales necessàries per als menús del portal:

```
(function($) {
  /**
   * Adds toggle link.
   * Toggles menu on small resolutions.
   * Restores menu on window width increasing.
   */
  Drupal.behaviors.responsiveBartikCollapsibleMenu = {
    attach: function (context, settings) {
      // We can keep menu collapsed up to width maxWidth.
      var maxWidth = 460;
      // Do nothing if menu is empty.
      if ($('#main-menu .menu a').length == 0) {
        return;
      }
      // Append toggle link to the main menu.
      $('#main-menu .navigation').append('<a href="#" id="menu-toggle">' +
      Drupal.t('Menu') + '</a>');
      // Collapse/expand menu by click on link.
      $('#a#menu-toggle').click(function() {
        $('#main-menu .menu').slideToggle('fast');
        return false;
      });
      // Restore visibility settings of menu on increasing of windows width over
```

460px.

// Media query works with width up to 460px. But I guess we should take into account some padding.

```
$(window).resize(function(){
    var w = $(window).width();
// Remove all styles if window size more than maxWidth and menu is hidden.
    if(w > maxWidth && $('#main-menu .menu').is(':hidden')) {
        $('#main-menu .menu').removeAttr('style');
    }
});
}
```

Aquestes etiquetes s'han afegit al final de l'arxiu *style.css* per a poder ser utilitzades un cop l'script canviï la pàgina. També s'inclou mostrar el bloc per a resolucions inferiors a 460 píxels i ocultar-lo per a resolucions superiors:

```
a#menu-toggle {
    display:none;
    padding: 0.7em 0 0.7em 10px;
    text-decoration: none;
    text-align: center;
    text-shadow: 0 1px #EEEEEE;
    position:relative;
    color: #000;
    background: rgba(255, 255, 255, 1);
    border-radius: 8px;
    margin-bottom: 5px;
    padding: 0.9em 5px;
}
a#menu-toggle:after {
    content:"";
    background: url('../images/toggle.png') no-repeat;
    width: 22px;
```



```

height: 30px;
display: inline-block;
position: absolute;
right: 10px;
}

@media all and (max-width: 460px) {
  a#menu-toggle {
    display: block;
  }
  [...]
}

@media all and (min-width: 461px) {
  a#menu-toggle {
    display: none;
  }
}

```

El resultat d'aquestes modificacions per a cada tipus de resolució es pot veure en la figura 34.

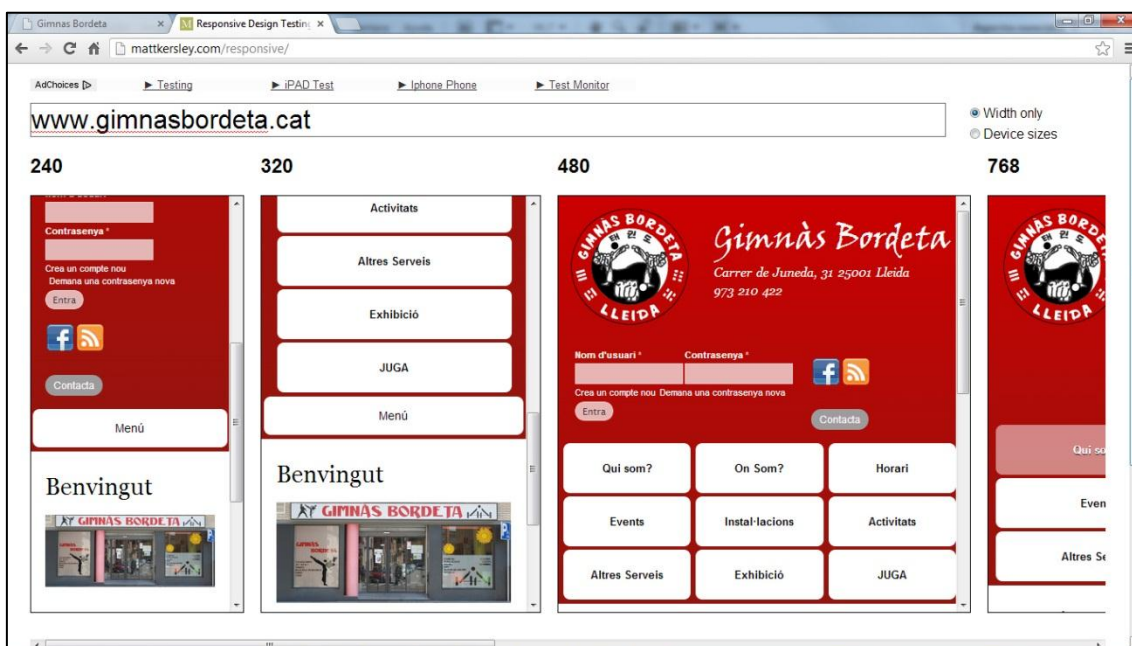


Figura 34: Responsiu

4.1.2 Mòduls d'autoria externa:

Algunes de les funcionalitats del portal s'han portat a terme

mitjançant l'ús de mòduls desenvolupats per altres usuaris que els han compartit de manera gratuïta a través de la pàgina web de *Drupal*.

4.1.2.1 Editor WYSIWYG:

WYSIWYG és l'acrònim de *What You See Is What You Get* (en català, "el que veus és el que obtens"). S'aplica als processadors de text i altres editors de text amb format (com els editors d'HTML, en aquest cas) que permeten escriure un document veient directament el resultat final, sense necessitat de conèixer el llenguatge en qüestió.

Per tal de que el personal administratiu i els usuaris en general puguin crear continguts nous d'una manera visual, s'utilitzen dos mòduls (*Insert* i *jwysiwyg*). El primer afegeix la possibilitat de crear tipus de contingut amb elements d'imatge que es pugen al servidor en comptes de a la base de dades (molt útil si no vols una base de dades de 700 Mb d'imatges). El segon substitueix (per als tipus d'usuaris seleccionat) el editor de textos predeterminat del sistema, afegint una senzilla interfície visual, que a més permet escalar imatges arrastrant amb el punter. A més es modifiquen els permisos dels usuaris per augmentar la seguretat. Únicament els administradors poden escriure en format *Full html*, mentre que els demés usuaris utilitzen el *Filtered html*. Això vol dir que els usuaris registrats no poden posar cap script, únicament imatges () i links(<a>).

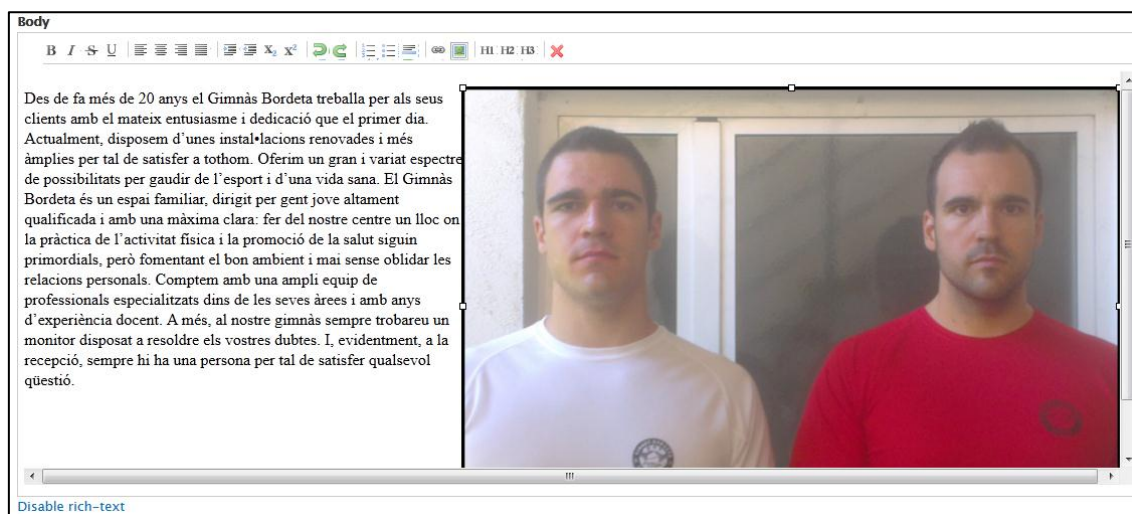


Figura 35: Editor de textos

4.1.2.2 Simple Gmaps:

Per a fer una pàgina de continguts de *Google maps* s’ha utilitzat el mòdul *Simple Google Maps*. Aquest mòdul requereix de la creació d’un tipus de contingut amb un camp de text anomenat *Adress*, però en les opcions de visualització es marca com a “*Google map from one line adress*”. El mòdul permet la configuració de la visualització predeterminada, el nivell de zoom i el tipus de mapa. El nivell de zoom és l’adequat, però el tipus de mapa es canvia a “Mapa de carrers”, per poder visualitzar direccions.

El valor d’aquest camp en la pàgina creada ha de ser igual a la recerca que es faria a *Google Maps*, en aquest cas “Gimnas Bordeta, Carrer de Juneda, Lleida”. El mateix buscador ja selecciona el local en qüestió.

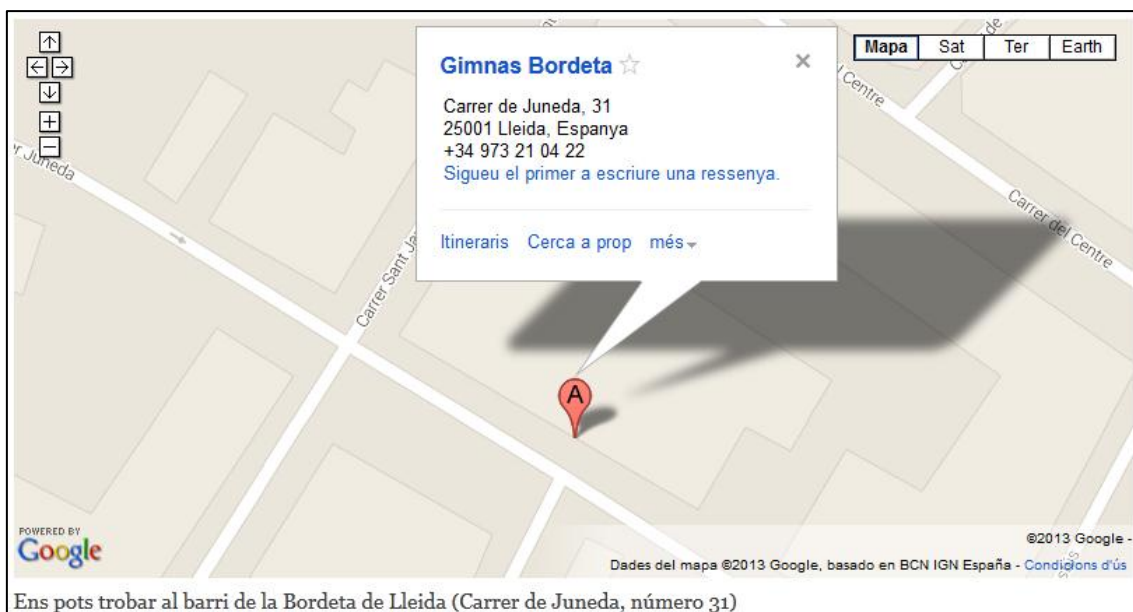


Figura 36: Simple GMaps

4.1.2.3 Slideshows:

Els *slideshows* o visualitzacions d’imatges lliscants, és una manera visualment atractiva de posar a la vista un grup de fotografies. Tot i que la majoria d’usuaris recomanen la creació de vistes mitjançant el mòdul *Views*, es troba que resulta massa complicat per a una tasca tant senzilla. Així doncs s’utilitza el mòdul *Dynamic Display Blocks*, també conegut com *DDBlock*, per tal d’integrar una passa de diapositives mitjançant *jQuery*

Cycle (que es té que descarregar a part des de la pàgina oficial).

Per crear un contingut per mostrar, des del menú d'administració "Estructura" fem clic en "*Dynamic display block*", on apareix un camp per crear un bloc, s'ingresa el nom de l'*slide* i es clica en *add block*. En *input type* es selecciona *Image folder* i s'indica on es troba aquest directori (per defecte *../images/ddblock*). Es canvia per *../images/[Nom del grup de imatges]* (aquest directori fa referència al directori definit a Configuració en el bloc de *MEDIA* a *File System*), ja que s'ha de crear un *slideshow* per a cada apartat, i cadascun contindrà les imatges en un subdirectorio diferent. S'especifica el nombre d'imatges i que s'utilitzarà un *imatge pager* (una petita mostra de la imatge) com a botó de seleccionar. Per últim s'especifica la duració de la transició entre les imatges i el temps de permanència de cadascuna.

Abans de crear el bloc, s'especifica que aquest només serà visible des d'un tipus de contingut que anomenarem *[nom del slide]main*. Així encara que compartiran ubicació, no apareixeran alhora. Es crea únicament una pàgina de cada tipus i es situen dins del Menú Principal mantenint l'estructura especificada.

En aquesta pàgina, es posa una breu descripció de l'activitat en el `<body>` així com una taula simple que mostra l'horari de l'activitat, damunt de l'*slideshow*.

Taekwondo

El "camí del peu i del puny"; art marcial sense armes on l'individu obté força, velocitat, resistència, flexibilitat, elasticitat, entre d'altres. A més, fomenta un valors que enforteixen l'esperit: integritat, perseverança, cortesia, autocontrol i l'esperit indòmit.

[Consulta](#)

Grups	Dilluns	Dimarts	Dimecres	Dijous	Divendres
Infantil	18:00 a 19:00		18:00 a 19:00		18:00 a 19:00
Juvenil	19:00 a 20:00		19:00 a 20:00		19:00 a 20:00
Adults	20:30 a 21:30		20:30 a 21:30		20:30 a 21:30






Figura 37: Slideshow

4.1.2.4 Calendari:

Inicialment es va preveure un calendari estàtic (una imatge), però degut a la natura dels tornejos els horaris de les activitats poden canviar entre setmanes. L'objectiu en l'apartat dels calendaris és realitzar un calendari dinàmic, que agafi dates des de tipus de contingut i els mostri en una graella. *Drupal 7* no accepta camps de data de manera nativa, així que es requerirà la interacció de tres mòduls diferenciats per a realitzar la tasca: *date* és un senzill mòdul que permet a drupal afegir i tractar camps com a data, incloent anys, dies, setmanes, mesos, minuts i hores. *Views* és un versàtil mòdul que permet la creació de "vistes" (pàgines creades dinàmicament en funció de les dades contingudes en altres pàgines). I per últim *Calendar*, que integra els dos mòduls anteriors i permet crear una vista de *views* d'elements *date*, en una graella en forma de calendari (tot i que fa més coses, aquesta és la funcionalitat que es desitja).

Com la interacció amb el mòdul *Views* pot ser un pel complicada, *Calendar* inclou un assistent al que es pot accedir des de Administració » Configuració » Date API » Date Wizard.

L'assistent guia en el procés de crear un tipus de contingut nou, a partir del qual el nou calendari adquirirà les dates. Dos tipus de contingut són creats,

per tal de tenir dos calendaris diferents: Un pels horaris del dia a dia, i un pels dies de les activitats especials en les que participa el centre.

Des d'Administració » Estructura » Views es veu que s'han creat dues noves vistes, amb quatre links per cadascuna depenent de les opcions de visualització (Dia, Setmana, Mes i Any). Els links de setmana per a les classes i de mes per a les activitats es situen en el Menú principal.

Per a cada assignatura, es crea una pàgina del tipus de contingut de les classes i s'utilitza el codi que presenta el portal (visual, amb un calendari on marcar els dies) els dies i hores en les que es repeteix, incloent la hora de finalització. El mateix per cada cop que hi ha una ocasió especial, però amb un element del tipus de contingut de les activitats, on a més es posa un breu resum de l'activitat per quan l'usuari s'interessi per ella.

El gran *handicap* d'aquests mòduls es que es troben totalment en anglès, i per tant les seves visualitzacions també, així doncs s'han traduït a castellà i català des de Administració » "Tradueix la Interfície", cadena a cadena.

Hora	dilluns	dimarts	dimecres	dijous	divendres	dissabte	diumenge
08:00	Sala de fitness 08/19/2013 - 08:00 fins 22:30	Sala de fitness 08/20/2013 - 08:00 fins 22:30	Sala de fitness 08/21/2013 - 08:00 fins 22:30	Sala de fitness 08/22/2013 - 08:00 fins 22:30	Sala de fitness 08/23/2013 - 08:00 fins 22:30		
09:30		Defensa Personal 08/20/2013 - 09:30 fins 11:00		Defensa Personal 08/22/2013 - 09:30 fins 11:00			
10:00						Sala de fitness 08/24/2013 - 10:00 fins 14:00	
11:00						Kick Boxing 08/24/2013 - 11:00 fins 12:30	
17:30					Yoga 08/23/2013 - 17:30 fins 18:30		

Figura 38: Horaris

Qui som?	On Som?	Horari	Events	Instal·lacions	Activitats	Altres Serveis	Exhibició	JUGA	Administrar
----------	---------	--------	--------	----------------	------------	----------------	-----------	------	-------------

Mes	Setmana	Dia	Any
-----	---------	-----	-----

Imprimir

agost 2013

« Prev Next »

dilluns	dimarts	dimecres	dijous	divendres	dissabte	diumenge
29	30	31	1 Activitats d'Estiu 08/01/2013 - 09:00 fins 13:00	2 Activitats d'Estiu 08/02/2013 - 09:00 fins 13:00	3	4
5	6	7	8	9	10	11
12	13	14	15	16 Vacances 08/16/2013 - 08:00 fins 08/31/2013 - 14:00	17 Vacances 08/16/2013 - 08:00 fins 08/31/2013 - 14:00	18 Vacances 08/16/2013 - 08:00 fins 08/31/2013 - 14:00
19 Vacances 08/16/2013 - 08:00 fins	20 Vacances 08/16/2013 - 08:00 fins	21 Vacances 08/16/2013 - 08:00 fins	22 Vacances 08/16/2013 - 08:00 fins	23 Vacances 08/16/2013 - 08:00 fins	24 Vacances 08/16/2013 - 08:00 fins	25 Vacances 08/16/2013 - 08:00 fins

Figura 39: Events

A més, en cas de que algun usuari vulgui imprimir, per a més comoditat s'ha afegit un petit script "Imprimir" a dalt a l'esquerra del calendari, que és el mateix que fer Arxiu » Imprimir. Això es té que posar en la capçalera de la vista:

```
<form>

<input type="button" name="imprimir" value="Imprimir"
onclick="window.print();">

</form>
```

4.1.2.5 Contacta:

La pàgina de contacte ja ve definida pel mòdul natiu de *Drupal* anomenat *Contact*. Simplement s'han creat dos comptes diferents on enviar els formularis: al webmaster (*website feedback*) o a Consultes Gimnàs Bordeta (Que ho reben els administradors del recinte). S'ha agregat un botó al bloc *Addthis.com* amb la direcció "*=?Contact*" que redirecciona al formulari. El fet de que s'hagi ubicat dins del bloc ve donat perquè es el bloc que més a la vista està, en totes les pàgines.



Formulari de contacte amb els camps següents:

- El vostre nom ***: Text "Tech"
- La vostra adreça electrònica ***: Text "jordi@virgili.org"
- Assumpte ***: Camp buit
- Categoria ***: Menú desplegable amb "Consultes Gimnàs Bordeta" seleccionat
- Missatge ***: Àrea de text gran amb un cursor al final
- ☐ Envieu-vos una còpia
- Botó: Envia el missatge

Figura 40: Formulari de contacte

4.1.2.6 Capcha:

Als primers dies de tenir el portal web funcionant, salta a la vista que part dels nous usuaris registrats resulten tenir un origen il·legítim (sobretot *spambots* després d’haver-se publicitat el portal a través de diverses xarxes socials). Tot i que està definit que els nous usuaris han de rebre l’aprovació per part d’un dels administradors al registrar-se (per tant aquests bots no tenen els comptes activats), es vol que encara menys bots arribin al filtre de l’administrador humà.

Es per això que s’integra el sistema de capcha (reconeixement d’imatge), amb la millora i ampliació *Re-Capcha* de *Google* per al registre d’usuaris.



Figura 41: Re-Capcha

Per tal d'habilitar el mòdul, es necessari crear compte a *Google Re-Capcha* (el més utilitzat del moment, i per tant el que la gent reconeixerà més fàcilment). Es demana una clau global a *Google* (<https://www.google.com/recaptcha/whyrecaptcha>) , i el servei retorna una clau publica i una privada (per realitzar una comunicació segura).

Amb el mòdul configurat, es fa aparèixer el *widget* de *ReCapcha* en el formulari de registre d'usuari del portal i com a resultat es nota un gran.

4.1.2.7 Encriptat de dades:

En el següent apartat és dissenyaran uns quants mòduls que desenvoluparan les funcionalitats restants de la pàgina. Tot i que els mètodes d'interacció amb la base de dades de *Drupal* és molt segur, es vol incrementar la seguretat en alguns de les dades que s'emmagatzemaran *AES encryption* és un senzill mòdul d'encriptació/desenciptació que permet l'ús de funcions d'encriptat en el codi *php*:

```
$encriptat = aes_encrypt("dades");

$desencriptat = aes_decrypt($encriptat);
```

La clau de l'encriptat pot estar ubicada en la base de dades o en un fitxer dins del servidor, per fer encara més segura l'aplicació. S'escull la encriptació més senzilla (*Rijndael 128*), ja que només es vol confondre en cas d'accés no autoritzat a la base de dades.

4.2 Disseny de mòduls:

Per poder completar la resta de funcionalitats projectades per a aquest projecte, s'han creat quatre mòduls. *Admin gym* i *Admin gym funcions extra*, són mòduls complementaris d'administració. *Gamifica* i *Mòdul Usuari* són mòduls complementaris per al projecte de gamificació.

I és que el mòdul de gamificació és la novetat que es proposa en aquest treball comparant amb altres portals i serveis disponibles actualment. El mòdul Gamifica, posa en comú els exercicis que realitzen els usuaris, generant classificacions que empenyen a superar-se entre ells. Complementant-lo, el Mòdul (de gamificació) Usuari permet crear fites o premis, i assignar-los als clients que assoleixin objectius. Aquests mòduls no són estàtics, ja que (fins i tot en el de fites) es permet als administradors (i en futures projeccions als usuaris mateixos) crear nous continguts al respecte, mantenint la tensió i l'interès dels participants.

4.2.1 Admin gym:

4.2.1.1 Info:

Aquest fitxer és el que determina els noms que rebrà el mòdul dins l'assistent de mòduls de *Drupal*. Inclou la descripció, el paquet dins del qual s'englobarà el mòdul (tots els creats per l'autor d'aquest projecte s'engloben dins del paquet "*Drupal 7 Virgili.org*"), i els fitxers que necessita el mòdul per a funcionar. En el fitxer *.install* hi ha les dades d'inicialització de les taules de la base de dades que requereix el mòdul, i en el fitxer *.module* les implementacions del codi que realitza les diferents tasques.

```
;$Id$  
name = Admin gym  
description = Modul per a gestionar asistencia i factures.  
package = Drupal 7 Virgili.org  
core = 7.x  
  
files[] = adm_gest.module  
files[] = adm_gest.install
```

4.2.1.2 Install:

Aquest fitxer inicialitza les taules dins de la base de dades que utilitzarà el mòdul per emmagatzemar dades. *Drupal 7* únicament utilitza l'anomenat *hook_schema()* al activar el mòdul i al desinstalar-lo (esborrant el contingut de totes les taules).

```
/**  
 * Implements hook_schema().  
 */
```

· La base de dades client guarda la informació administrativa del mateix. Com la matrícula també relaciona les activitats i el NIF pot ser el del pare, que també es membre, no es poden utilitzar com a camps clau.

```
function adm_gest_schema() {  
  $schema['client'] = array(  
    'description' => 'client del gimnas.',  
    'fields' => array(  
      'num' => array(  
        'description' => 'numero de la entrada.',  
        'type' => 'serial',  
        'unsigned' => TRUE,  
        'not null' => TRUE,  
      ),  
      'matricula' => array(  
        'description' => 'numero de matricula del participant.',  
        'type' => 'varchar',  
        'length' => 6,  
        'not null' => TRUE,  
        'default' => 0,  
      ),  
      'nif' => array(  
        'description' => 'document identitat.',  
        'type' => 'varchar',  
        'length' => 10,
```

```
'not null' => TRUE,
'default' => 0,
),
'nom' => array(
  'description' => 'nom del client.',
  'type' => 'varchar',
  'length' => 12,
  'not null' => TRUE,
  'default' => 0,
),
'cognom1' => array(
  'description' => 'primer cognom.',
  'type' => 'varchar',
  'length' => 16,
  'not null' => TRUE,
  'default' => 0,
),
'cognom2' => array(
  'description' => 'segon cognom.',
  'type' => 'varchar',
  'length' => 16,
  'not null' => TRUE,
  'default' => 0,
),
'email' => array(
  'description' => 'e-mail de contacte.',
  'type' => 'varchar',
  'length' => 64,
  'not null' => TRUE,
  'default' => 0,
),
'telefix' => array(
  'description' => 'telefon fix.',
  'type' => 'varchar',
```

```
'length' => 64,
'not null' => TRUE,
'default' => 0,
),
'telemob' => array(
  'description' => 'telefon mobil.',
  'type' => 'varchar',
  'length' => 64,
  'not null' => TRUE,
  'default' => 0,
),
'ciutat' => array(
  'description' => 'ciutat de residencia.',
  'type' => 'varchar',
  'length' => 16,
  'not null' => TRUE,
  'default' => 0,
),
'adresa' => array(
  'description' => 'adreça de residencia.',
  'type' => 'varchar',
  'length' => 64,
  'not null' => TRUE,
  'default' => 0,
),
'dnaix' => array(
  'description' => 'dia de naixement.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'mnaix' => array(
  'description' => 'mes de naixement.',
```

```
'type' => 'int',
'size' => 'small',
'not null' => TRUE,
'default' => 0,
),
'anaix' => array(
  'description' => 'any de naixement.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'dmat' => array(
  'description' => 'dia de matriculacio.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'mmat' => array(
  'description' => 'mes de matriculacio.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'amat' => array(
  'description' => 'any de matriculacio.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'compaga' => array(
  'description' => 'si utilitza domiciliacio, especificar
```



```
        compte.',
        'type' => 'varchar',
        'length' => 64,
        'not null' => TRUE,
        'default' => 0,
    ),
    'coment' => array(
        'description' => 'Comentaris inscripcio.',
        'type' => 'varchar',
        'length' => 256,
        'not null' => TRUE,
        'default' => 'Cap comentari.',
    ),
),
'primary key' => array('num'),
);
```

- Asistencia serveix per a emmagatzemar les dades de assistència d'un client, associada a un dia i a una hora.

```
$schema['asistencia'] = array(
    'description' => 'Us dels diferents equipaments.',
    'fields' => array(
        'num' => array(
            'description' => 'Numero de la entrada.',
            'type' => 'serial',
            'unsigned' => TRUE,
            'not null' => TRUE,
        ),
        'nom' => array(
            'description' => 'Nom del client.',
            'type' => 'varchar',
            'length' => 12,
            'not null' => TRUE,
            'default' => 0,
        ),
    ),
);
```

```
'cognom1' => array(
  'description' => 'Primer cognom.',
  'type' => 'varchar',
  'length' => 16,
  'not null' => TRUE,
  'default' => 0,
),
'cognom2' => array(
  'description' => 'segon cognom.',
  'type' => 'varchar',
  'length' => 16,
  'not null' => TRUE,
  'default' => 0,
),
'hass' => array(
  'description' => 'hora asistencia.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'mmass' => array(
  'description' => 'minut asistencia.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'dass' => array(
  'description' => 'dia asistencia.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
```

```

    ),
    'mass' => array(
        'description' => 'mes asistencia.',
        'type' => 'int',
        'size' => 'small',
        'not null' => TRUE,
        'default' => 0,
    ),
    'aass' => array(
        'description' => 'any asistencia.',
        'type' => 'int',
        'size' => 'small',
        'not null' => TRUE,
        'default' => 0,
    ),
),
),
'primary key' => array('num'),
);

```

- La taula taekwondo guarda les dades de federació de taekwondo associades a un client, en cas d'haver-n'hi.

```

$schema['taekwondo'] = array(
    'description' => 'Usuaris de competicio.',
    'fields' => array(
        'num' => array(
            'description' => 'Numero de la entrada.',
            'type' => 'serial',
            'unsigned' => TRUE,
            'not null' => TRUE,
        ),
        'nif' => array(
            'description' => 'document identitat.',
            'type' => 'varchar',
            'length' => 10,
            'not null' => TRUE,
        ),
    ),
);

```

```
        'default' => 0,
    ),
    'federacio' => array(
        'description' => 'Numero de federat.',
        'type' => 'varchar',
        'length' => 6,
        'not null' => TRUE,
        'default' => 0,
    ),
    'dfed' => array(
        'description' => 'dia de federacio.',
        'type' => 'int',
        'size' => 'small',
        'not null' => TRUE,
        'default' => 0,
    ),
    'mfed' => array(
        'description' => 'mes de federacio.',
        'type' => 'int',
        'size' => 'small',
        'not null' => TRUE,
        'default' => 0,
    ),
    'afed' => array(
        'description' => 'any de federacio.',
        'type' => 'int',
        'size' => 'small',
        'not null' => TRUE,
        'default' => 0,
    ),
    ),
    'primary key' => array('num'),
);
```

- La taula cinturo guarda les dades de cinturons de taekwondo associades a

un client de taekwondo, en cas d'haver-n'hi.

```
$schema['cinturo'] = array(  
  'description' => 'Cinturons.',  
  'fields' => array(  
    'num' => array(  
      'description' => 'Numero de la entrada.',  
      'type' => 'serial',  
      'unsigned' => TRUE,  
      'not null' => TRUE,  
    ),  
    'nif' => array(  
      'description' => 'document identitat.',  
      'type' => 'varchar',  
      'length' => 10,  
      'not null' => TRUE,  
      'default' => 0,  
    ),  
    'color' => array(  
      'description' => 'color del cinturo.',  
      'type' => 'varchar',  
      'length' => 12,  
      'not null' => TRUE,  
      'default' => 0,  
    ),  
    'dcin' => array(  
      'description' => 'dia de cinturo.',  
      'type' => 'int',  
      'size' => 'small',  
      'not null' => TRUE,  
      'default' => 0,  
    ),  
    'mcin' => array(  
      'description' => 'mes de cinturo.',  
      'type' => 'int',
```

```
        'size' => 'small',
        'not null' => TRUE,
        'default' => 0,
    ),
    'acin' => array(
        'description' => 'any de cinturó.',
        'type' => 'int',
        'size' => 'small',
        'not null' => TRUE,
        'default' => 0,
    ),
),
'primary key' => array('num'),
);
```

- La taula banc guarda les gestions bancàries associades a un client, en cas d'haver-n'hi, per generar un historial de pagaments.

```
$schema['banc'] = array(
    'description' => 'Dades banc.',
    'fields' => array(
        'num' => array(
            'description' => 'Número de la entrada.',
            'type' => 'serial',
            'unsigned' => TRUE,
            'not null' => TRUE,
        ),
        'nif' => array(
            'description' => 'document identitat.',
            'type' => 'varchar',
            'length' => 10,
            'not null' => TRUE,
            'default' => 0,
        ),
        'import' => array(
```

```
'description' => 'import cobrat.',
'type' => 'int',
'size' => 'small',
'not null' => TRUE,
'default' => 0,
),
'dpag' => array(
  'description' => 'dia de cobrament.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'mpag' => array(
  'description' => 'mes de cobrament.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
'apag' => array(
  'description' => 'any de cobrament.',
  'type' => 'int',
  'size' => 'small',
  'not null' => TRUE,
  'default' => 0,
),
),
'primary key' => array('num'),
);

return $schema;
}
```


4.2.1.3 Module:

En aquest fitxer es pot trobar la implementació de les pàgines interactives d'administració. El fitxer comença amb un menú on podem veure les diferents opcions (*[nom_del_mòdul]_menu()*) i on redirecciona cadascuna (*\$items['url']*).

```
function adm_gest_menu() {
    $items = array();
    $items['adm_gest/intro'] = array(
        'title' => t('Introduir nova gent en el sistema'),
        'page callback' => 'adm_gest_intro',
        'access arguments' => array('access content'),
        'description' => t('Nous clients'),
        'type' => MENU_CALLBACK,
    );
    $items['adm_gest/asist'] = array(
        'title' => t('Assistència'),
        'page callback' => 'adm_gest_asist',
        'access arguments' => array('access content'),
        'description' => t('Resultats'),
        'type' => MENU_CALLBACK,
    );
    $items['adm_gest/taekwo'] = array(
        'title' => t('Gestoinar usuari de competició'),
        'page callback' => 'adm_gest_taekwo',
        'access arguments' => array('access content'),
        'description' => t('Taekwondo'),
        'type' => MENU_CALLBACK,
    );
    $items['adm_gest/banc'] = array(
        'title' => t('Gestoinar pagaments'),
        'page callback' => 'adm_gest_banc',
        'access arguments' => array('access content'),
        'description' => t('Gestió pagaments'),
    );
}
```

```

    'type' => MENU_CALLBACK,
  );
  $items['adm_gest/usuari'] = array(
    'title' => t('Informació Usuari'),
    'page callback' => 'adm_gest_usuari',
    'access arguments' => array('access content'),
    'description' => t('Visualitzar Usuari'),
    'type' => MENU_CALLBACK,
  );
  return $items;
}

```

4.2.1.3.1 Nous clients

- El primer dels menús porta a la introducció de nova gent en el sistema. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function adm_gest_form_submit($form_id, $form_values) {
```

- En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

```

if ($form_values['values']['passwd'] != 'antilistos'){
  return drupal_set_message(t('Segur? jo crec que el password no es el que toca...'));
}

```

- Si pot continuar, mostra els camps del formulari per pantalla.

```

drupal_set_message(t('Matrícula: %string.', array('%string' =>
$form_values['values']['matricula'])));

drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom'])));

drupal_set_message(t('Primer cognom: %string.', array('%string' =>
$form_values['values']['cognom1'])));

drupal_set_message(t('Segon cognom: %string.', array('%string' =>
$form_values['values']['cognom2'])));

drupal_set_message(t('NIF o NIE %string.', array('%string' =>
$form_values['values']['nif'])));

drupal_set_message(t('Telèfon fix: %string.', array('%string' =>
$form_values['values']['telfix'])));

```

```

drupal_set_message(t('Telèfon Mòbil: %string.', array('%string' =>
$form_values['values']['telmob'])));

drupal_set_message(t('E-mail: %string.', array('%string' =>
$form_values['values']['email'])));

drupal_set_message(t('Adreça: %string.', array('%string' =>
$form_values['values']['adresa'])));

drupal_set_message(t('Localitat: %string.', array('%string' =>
$form_values['values']['ciutat'])));

drupal_set_message(t('Naixement:'));

drupal_set_message(t('Dia %string.', array('%string' =>
$form_values['values']['naixdia']['day'])));

drupal_set_message(t('Mes %string.', array('%string' =>
$form_values['values']['naixdia']['month'])));

drupal_set_message(t('Any %string.', array('%string' =>
$form_values['values']['naixdia']['year'])));

drupal_set_message(t('Matriculació:'));

drupal_set_message(t('Dia %string.', array('%string' =>
$form_values['values']['matdia']['day'])));

drupal_set_message(t('Mes %string.', array('%string' =>
$form_values['values']['matdia']['month'])));

drupal_set_message(t('Any %string.', array('%string' =>
$form_values['values']['matdia']['year'])));

drupal_set_message(t('Com paga: %string.', array('%string' =>
$form_values['values']['compaga'])));

drupal_set_message(t('Comentaris: %string.', array('%string' =>
$form_values['values']['comment'])));

```

· Comprova si l'usuari ja existeix en la taula client.

```

$query = db_select('client');

$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
    ->condition('nom', $form_values['values']['nom'], '=')
    ->condition('cognom1', $form_values['values']['cognom1'], '=')
    ->condition('cognom2', $form_values['values']['cognom2'], '=');

$result = $query->execute();

$record = $result->fetchAssoc();

```

· En cas d'existir, informa i es para.

```

if ($record['nif']!=NULL){

return drupal_set_message(t('Aquest usuari ja existeix en la base de dades de
clients.'), 'error');

}

```

- Encripta les dades sensibles (telèfons, adreça i forma de pagament (o compte corrent)).

```
$tf = aes_encrypt($form_values['values']['telfix']);
$tm = aes_encrypt($form_values['values']['telmob']);
$adress = aes_encrypt($form_values['values']['adresa']);
$paga = aes_encrypt($form_values['values']['compaga']);
```

- Si l'usuari té rol d'administrador, inclou al nou usuari en la base de dades.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$posar = db_insert('client')
->fields(array(
    'nom' => $form_values['values']['nom'],
    'cognom1' => $form_values['values']['cognom1'],
    'cognom2' => $form_values['values']['cognom2'],
    'email' => $form_values['values']['email'],
    'telefix' => $tf,
    'telemob' => $tm,
    'matricula' => $form_values['values']['matricula'],
    'nif' => $form_values['values']['nif'],
    'ciutat' => $form_values['values']['ciutat'],
    'adresa' => $adress,
    'dnaix' => $form_values['values']['naixdia']['day'],
    'mnaix' => $form_values['values']['naixdia']['month'],
    'anaix' => $form_values['values']['naixdia']['year'],
    'dmat' => $form_values['values']['matdia']['day'],
    'mmat' => $form_values['values']['matdia']['month'],
    'amat' => $form_values['values']['matdia']['year'],
    'compaga' => $paga,
    'coment' => $form_values['values']['comment'],
))
->execute();
}

drupal_set_message(t('Usuari actualitzat correctament.'));
```

```
}
```

4.2.1.3.2 Asistència

· Aquest menú serveix per entrar dades en la taula d'Asistència en entrar un usuari al recinte. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function diari_form_submit($form_id, $form_values) {
```

· En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

```
if ($form_values['values']['passwd']!= 'asist13'){
return drupal_set_message(t('Segur? jo crec que el password no es el que
toca...'));
}
```

· Mostra els resultats del formulari per pantalla.

```
drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom'])));

drupal_set_message(t('Primer cognom: %string.', array('%string' =>
$form_values['values']['cognom1'])));

drupal_set_message(t('Nova entrada:'));

drupal_set_message(t('hora %string.', array('%string' =>
$form_values['values']['hour'])));

drupal_set_message(t('minuts %string.', array('%string' =>
$form_values['values']['minut'])));

drupal_set_message(t('Dia %string.', array('%string' =>
$form_values['values']['assdia']['day'])));

drupal_set_message(t('Mes %string.', array('%string' =>
$form_values['values']['assdia']['month'])));

drupal_set_message(t('Any %string.', array('%string' =>
$form_values['values']['assdia']['year'])));
```

· Insereix les dades de l'usuari en el camp assistència.

```
$posar = db_insert('asistencia')
```

```

->fields(array(
    'nom' => $form_values['values']['nom'],
    'cognom1' => $form_values['values']['cognom1'],
    'cognom2' => $form_values['values']['cognom2'],
    'hass' => $form_values['values']['hour'],
    'mmass' => $form_values['values']['minut'],
    'dass' => $form_values['values']['assdia']['day'],
    'mass' => $form_values['values']['assdia']['month'],
    'aass' => $form_values['values']['assdia']['year'],
))
->execute();

drupal_set_message(t('Entrada actualitzada correctament.'));

```

- Comprova si aquest client existeix en la base de dades de clients.

```

$query = db_select('client');

$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
    ->condition('nom', $form_values['values']['nom'], '=')
    ->condition('cognom1', $form_values['values']['cognom1'], '=')
    ->condition('cognom2', $form_values['values']['cognom2'], '=');

$result = $query->execute();

$record = $result->fetchAssoc();

$mes=date('n');

$any=date('Y');

```

- Es comprova si hi ha resultats en la taula de pagaments per al mes actual de l'any actual.

```

$query4 = db_select('banc');

$query4->fields('banc', array('nif', 'import', 'mpag', 'apag'))
    ->condition('nif', $record['nif'], '=')
    ->condition('mpag', $mes, '=')

```

```
->condition('apag', $any, '=');
```

```
$result4 = $query4->execute();
```

```
$record4 = $result4->fetchAssoc();
```

- Es mostra per pantalla un missatge informant a l'administrador si ha pagat o encara no.

```
if ($record4['nif']==NULL){
```

```
    drupal_set_message(t('Aquest usuari no ha pagat aquest mes.'), 'error');
```

```
}
```

```
if ($record4['nif']!=NULL){
```

```
    drupal_set_message(t('Aquest usuari ha pagat aquest mes.'));
```

```
}
```

```
}
```

4.2.1.3.3 Gestionar usuari de competició

- Aquest element serveix per afegir dades de la federació de Taekwondo per a un client. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function taekwo_form_submit($form_id, $form_values) {
```

- En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

```
if ($form_values['values']['passwd']!='antilistos'){
```

```
    return drupal_set_message(t('Segur? jo crec que el password no es el que toca...'));
```

```
}
```

- Mostra els valors introduïts en el formulari per pantalla.

```
drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom'])));
```

```
drupal_set_message(t('Primer cognom: %string.', array('%string' =>
$form_values['values']['cognom1'])));
```

```
drupal_set_message(t('Segon cognom: %string.', array('%string' =>
$form_values['values']['cognom2'])));
```

```
drupal_set_message(t('Número de federat: %string.', array('%string' =>
```

```

$form_values['values']['federat']));

drupal_set_message(t('Data federació: %string.', array('%string' =>
$form_values['values']['tkdia']['day'].'/'.$form_values['values']['tkdia']['month'].'/'.$form_values['values']['tkdia']['year'])));

drupal_set_message(t('Color cinturó: %string.', array('%string' =>
$form_values['values']['cintcol'])));

drupal_set_message(t('Data cinturó: %string.', array('%string' =>
$form_values['values']['cindia']['day'].'/'.$form_values['values']['cindia']['month'].'/'.$form_values['values']['cindia']['year'])));

```

- Comprova si el client existeix en la base de dades de clients.

```

$query = db_select('client');

$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
    ->condition('nom', $form_values['values']['nom'], '=')
    ->condition('cognom1', $form_values['values']['cognom1'], '=')
    ->condition('cognom2', $form_values['values']['cognom2'], '=');

$result = $query->execute();

$record = $result->fetchAssoc();

```

- En cas de no existir, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```

if ($record['nif']==NULL){

    drupal_set_message(t('Aquest usuari no existeix en la base de dades de clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {

        $query0 = db_select('client');

        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['cognom1'], '=');

        $result0 = $query0->execute();

        $record0 = $result0->fetchAssoc();

        if ($record0['nif']!=NULL){

            drupal_set_message(t('Possibles clients amb aquest primer cognom en color verd.'), 'error');

            drupal_set_message(t('Nom: %string.', array('%string' =>
            $record0['nom'].' '.$record0['cognom1'].' '.$record0['cognom2'])));

            while($record0 = $result0->fetchAssoc()){

                drupal_set_message(t('Nom: %string.', array('%string' =>
                $record0['nom'].' '.$record0['cognom1'].' '.$record0['cognom2'])));
            }
        }
    }
}

```



```

    }

    }

    return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');

    }

return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
'error');

}

drupal_set_message(t('nif: %string.', array('%string' => $record['nif'])));

```

· Si l'usuari té permisos d'administrador, introdueix les dades en les taules corresponents i informa que la tasca s'ha realitzat.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {
$posar1 = db_insert('taekwondo')
->fields(array(
    'nif' => $record['nif'],
    'federacio' => $form_values['values']['federat'],
    'dfed' => $form_values['values']['tkdia']['day'],
    'mfed' => $form_values['values']['tkdia']['month'],
    'afed' => $form_values['values']['tkdia']['year'],
))
->execute();

$posar2 = db_insert('cinturo')
->fields(array(
    'nif' => $record['nif'],
    'color' => $form_values['values']['cintcol'],
    'dcin' => $form_values['values']['cindia']['day'],
    'mcin' => $form_values['values']['cindia']['month'],
    'acin' => $form_values['values']['cindia']['year'],
))
->execute();
}

```

```
drupal_set_message(t('Entrada actualitzada correctament.'));
}
```

4.2.1.3.4 Gestionar pagament

· Aquest element serveix per associar pagaments en una data a un client. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function banc_form_submit($form_id, $form_values) {
```

· En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

```
if ($form_values['values']['passwd'] != 'antilistos'){
return drupal_set_message(t('Segur? jo crec que el password no es el que
toca...'));
}
```

· Mostra els valors del formulari per pantalla.

```
drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom'])));

drupal_set_message(t('Primer cognom: %string.', array('%string' =>
$form_values['values']['cognom1'])));

drupal_set_message(t('Segon cognom: %string.', array('%string' =>
$form_values['values']['cognom2'])));

drupal_set_message(t('Import: %string.', array('%string' =>
$form_values['values']['import'])));

drupal_set_message(t('Transacció: %string.', array('%string' =>
$form_values['values']['pagadia']['day'].'/'.$form_values['values']['pagadia']
['month'].'/'.$form_values['values']['pagadia']['year'])));
```

· Si l'usuari té permisos d'administrador, comprova si existeix el client en la taula de clients.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');

$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
->condition('nom', $form_values['values']['nom'], '=')
->condition('cognom1', $form_values['values']['cognom1'], '=')
->condition('cognom2', $form_values['values']['cognom2'], '=');

$result = $query->execute();
```

```
$record = $result->fetchAssoc();
}
```

· En cas de no existir, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```
if ($record['nif']==NULL){
    drupal_set_message(t('Aquest usuari no existeix en la base de dades de
clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query0 = db_select('client');
        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['cognom1'],'=');

        $result0 = $query0->execute();
        $record0 = $result0->fetchAssoc();
        if ($record0['nif']!=NULL){
            drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');
            drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom'].' '.$record0['cognom1'].' '.$record0['cognom2'])));
            while($record0 = $result0->fetchAssoc()){
                drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom'].' '.$record0['cognom1'].' '.$record0['cognom2'])));
            }
        }

        return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');
    }

    return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
'error');
}

if (in_array('administrator', $GLOBALS['user']->roles)) {
    drupal_set_message(t('nif client: %string.', array('%string' =>
```

```
$record['nif']]]));
```

- En cas d'existir, introdueix els valors de la data i import de pagament, associat al NIF o NIE del client. Informa de que s'ha dut a terme la tasca.

```
$posar = db_insert('banc')
->fields(array(
  'nif' => $record['nif'],
  'import' => $form_values['values']['import'],
  'dpag' => $form_values['values']['pagadia']['day'],
  'mpag' => $form_values['values']['pagadia']['month'],
  'apag' => $form_values['values']['pagadia']['year'],
))
->execute();
}
drupal_set_message(t('Entrada actualitzada correctament.'));
}
```

4.2.1.3.5 Informació usuari

- Aquest element ens permet recaptar tota la informació de la que es disposa d'un usuari d'una manera senzilla i ràpida. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function usuari_form_submit($form_id, $form_values) {
  · En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

  if ($form_values['values']['passwd']!= 'antilistos'){
    return drupal_set_message(t('Segur? jo crec que el password no es el que toca...'));
  }

  $mes=date('n');
  $any=date('Y');
```

- Mostra per pantalla els valors del formulari.

```
//drupal_set_message(t('Nom: %string.', array('%string' =>
```

```
$form_values['values']['nom'])););

//drupal_set_message(t('Primer cognom: %string.', array('%string' =>
$form_values['values']['cognom1'])););

//drupal_set_message(t('Segon cognom: %string.', array('%string' =>
$form_values['values']['cognom2'])););
```

- Consulta la informació de la taula client.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');

$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif',
'matricula', 'email', 'telefix', 'telemob', 'ciutat', 'adresa', 'dnaix',
'mnaix', 'anaix', 'dmat', 'mmat', 'amat', 'compaga', 'coment'));

    ->condition('nom', $form_values['values']['nom'],'=')
    ->condition('cognom1', $form_values['values']['cognom1'],'=')
    ->condition('cognom2', $form_values['values']['cognom2'],'=');

$result = $query->execute();
$record = $result->fetchAssoc();
}
```

- En cas de no existir, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```
if ($record['nif']==NULL){
    drupal_set_message(t('Aquest usuari no existeix en la base de dades de
clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query0 = db_select('client');
        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['cognom1'],'=');

        $result0 = $query0->execute();
        $record0 = $result0->fetchAssoc();

        if ($record0['nif']!=NULL){
            drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');

            drupal_set_message(t('Nom: %string.', array('%string' =>
```

```

$record0['nom']. ' '.$record0['cognom1']. ' '.$record0['cognom2']));
    while($record0 = $result0->fetchAssoc()){
        drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '.$record0['cognom1']. ' '.$record0['cognom2']));
    }
}

return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');

}

return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
'error');

}

```

· Si existeix, descripta les dades sensibles i les guarda en variables temporals.

```

$tf = aes_decrypt($record['telefonix']);
$tm = aes_decrypt($record['telefonmob']);
$adress = aes_decrypt($record['adreça']);
$paga = aes_decrypt($record['compaga']);

```

· Mostra totes les dades de la taula client associades a l'usuari seleccionat.

```

drupal_set_message(t('Matrícula: %string.', array('%string' =>
$record['matricula']));

drupal_set_message(t('Data Matrícula: %string.', array('%string' =>
$record['dmat']. '/' . $record['mmat']. '/' . $record['amat']));

drupal_set_message(t('Nom: %string.', array('%string' => $record['nom'].
' '.$record['cognom1']. ' '.$record['cognom2']));

drupal_set_message(t('Data Naixement: %string.', array('%string' =>
$record['dnaix']. '/' . $record['mnaix']. '/' . $record['anaix']));

drupal_set_message(t('NIF o NIE: %string.', array('%string' =>
$record['nif']));

drupal_set_message(t('e-mail: %string.', array('%string' =>
$record['email']));

drupal_set_message(t('Telefon fix: %string.', array('%string' => $tf));
drupal_set_message(t('Telefon mòbil: %string.', array('%string' => $tm));
drupal_set_message(t('Adreça: %string.', array('%string' => $adress));
drupal_set_message(t('Localitat: %string.', array('%string' =>
$record['ciutat']));

drupal_set_message(t('Paga des de: %string.', array('%string' => $paga));
drupal_set_message(t('Comentari: %string.', array('%string' =>

```

```
$record['coment']));
```

- Comprova si hi ha dades de la federació de taekwondo associades.

```
$query2 = db_select('taekwondo');
```

```
$query2->fields('taekwondo', array('nif', 'federacio', 'dfed', 'mfed', 'afed'))
```

```
->condition('nif', $record['nif'],'=')
```

```
->orderBy('afed', 'DESC')
```

```
->orderBy('mfed', 'DESC');
```

```
$result2 = $query2->execute();
```

```
$record2 = $result2->fetchAssoc();
```

- Si en té, imprimeix per pantalla el numero de federació i la data de caducitat més nova, així com els cinturons del client.

```
if ($record2['nif']!=NULL){
```

```
    drupal_set_message(t('Número de federació de Taekwondo: %string.', array('%string' => $record2['federacio'])));
```

```
    drupal_set_message(t('Data de Caducitat: %string.', array('%string' => $record2['dfed'].'/'.$record2['mfed'].'/'.$record2['afed'])));
```

```
if ($record2['mfed'] < $mes && $record2['afed'] < $any){
```

```
    drupal_set_message(t('Llicència de taekwondo caducada.'), 'error');
```

```
}
```

```
    $query3 = db_select('cinturo');
```

```
    $query3->fields('cinturo', array('nif', 'color', 'dcin', 'mcin', 'acin'))
```

```
->condition('nif', $record['nif'],'=')
```

```
->orderBy('acin', 'DESC')
```

```
->orderBy('mcin', 'DESC');
```

```
$result3 = $query3->execute();
```

```
while ($record3 = $result3->fetchAssoc()){
```

```
    drupal_set_message(t('Color %string.', array('%string' => $record3['color'].' des del '.$record3['dcin'].'/'.$record3['mcin'].'/'.$record3['acin'])));
```

```
}
```

```
}
```

- Comprova les dades bancàries del client i si ha pagat aquest mes.

```
$query4 = db_select('banc');
$query4->fields('banc', array('nif', 'import', 'mpag', 'apag'))
    ->condition('nif', $record['nif'],'=')
    ->condition('mpag', $mes,'=')
    ->condition('apag', $any,'=');
$result4 = $query4->execute();
$record4 = $result4->fetchAssoc();
```

- En cas de no haver pagat, mostra un missatge d'error informant de la qüestió. Si ha pagat informa de que el client ja ha pagat aquest mes.

```
if ($record4['nif']==NULL){
    drupal_set_message(t('Aquest usuari no ha pagat aquest mes.'), 'error');
}
if ($record4['nif']!=NULL){
    drupal_set_message(t('Aquest usuari ha pagat aquest mes.'));
}
}
```

4.2.2 Admin gym funcions extra:

4.2.2.1 Info:

Aquest mòdul es un complement de l'anterior. Com a tal, s'estableixen dependències amb ell. També es veu que, al ser una expansió de les funcions de l'anterior, no conté cap fitxer amb extensió *.install*, ja que utilitza les taules del mòdul anterior.

```
name = Admin gym funcions extra
description = Segona part de area administrativa.
dependencies[] = adm_gest
package = Drupal 7 Virgili.org
core = 7.x
files[] = adm_gest_add.module
```


4.2.2.2 Module:

En aquest fitxer es pot trobar la expansió i implementació de les pàgines interactives d'administració. El fitxer comença amb un menú on podem veure les diferents opcions (*[nom_del_mòdul]_menu()*) i on redirecciona cadascuna (*\$items['url']*).

```
function adm_gest_add_menu() {
    $items = array();
    $items['adm/cintur'] = array(
        'title' => t('Actualitzar Cinturó'),
        'page callback' => 'adm_cint',
        'access arguments' => array('access content'),
        'description' => t('Actualitzar Cinturó'),
        'type' => MENU_CALLBACK,
    );
    $items['adm/federa'] = array(
        'title' => t('Actualitzar Dades Federació'),
        'page callback' => 'adm_federa',
        'access arguments' => array('access content'),
        'description' => t('Actualitza dades'),
        'type' => MENU_CALLBACK,
    );
    $items['adm/visubanc'] = array(
        'title' => t('Consultar dades Bancàries'),
        'page callback' => 'adm_visubanc',
        'access arguments' => array('access content'),
        'description' => t('Dades Bancàries'),
        'type' => MENU_CALLBACK,
    );
    $items['resultatbanca'] = array(
        'title' => t('Retorn dades Bancàries'),
        'page callback' => 'adm_resubanc',
        'access arguments' => array('access content'),
```

```
'description' => t('Retorn dades Bancàries'),
'type' => MENU_CALLBACK,
);
return $items;
}
```

4.2.2.2.1 Actualitzar cinturó

· Aquest element serveix per a afegir una nova associació de cinturó de taekwondo amb un client, que no té perquè ser una cosa amb data del mateix dia que s'envien les dades. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi.

```
function adm_cint_form_submit($form_id, $form_values) {
```

· En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

```
if ($form_values['values']['passwd']!='antilistos'){
return drupal_set_message(t('Segur? jo crec que el password no es el que
toca...'));
}
```

· Mostra un missatge amb les dades del formulari.

```
drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom'])));

drupal_set_message(t('Primer cognom: %string.', array('%string' =>
$form_values['values']['cognom1'])));

drupal_set_message(t('Segon cognom: %string.', array('%string' =>
$form_values['values']['cognom2'])));

drupal_set_message(t('Color cinturó: %string.', array('%string' =>
$form_values['values']['cintcol'])));

drupal_set_message(t('Data cinturó: %string.', array('%string' =>
$form_values['values']['cindia']['day'].'/'.$form_values['values']['cindia']['month'].'/'.$form_values['values']['cindia']['year'])));
```

· En cas de que l'usuari tingui permisos d'administrador, es comprova que el client existeixi en la taula clients.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');
$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
->condition('nom', $form_values['values']['nom'], '=')
->condition('cognom1', $form_values['values']['cognom1'], '=')
```

```

->condition('cognom2', $form_values['values']['cognom2'],'=');
$result = $query->execute();
$record = $result->fetchAssoc();
}

```

· En cas de no existir, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```

if ($record['nif']==NULL){

drupal_set_message(t('Aquest usuari no existeix en la base de dades de clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {

        $query0 = db_select('client');

        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))

            ->condition('cognom1', $form_values['values']['cognom1'],'=');

        $result0 = $query0->execute();

        $record0 = $result0->fetchAssoc();

        if ($record0['nif']!=NULL){

            drupal_set_message(t('Possibles clients amb aquest primer cognom en color verd.'),
            'error');

            drupal_set_message(t('Nom: %string.', array('%string' => $record0['nom'].
            '$record0[cognom1'].'.$record0[cognom2'])));

            while($record0 = $result0->fetchAssoc()){

                drupal_set_message(t('Nom: %string.', array('%string' =>
                $record0['nom'].'.$record0[cognom1'].'.$record0[cognom2'])));

            }

        }

        return drupal_set_message(t('Torni a realitzar la consulta.'), 'error');

    }
}

```

```
return drupal_set_message(t('Cap usuari amb aquest primer cognom.'), 'error');
}
```

```
drupal_set_message(t('nif: %string.', array('%string' => $record['nif'])));
```

· En cas de que l'usuari tingui permisos d'administrador, s'introdueixen les dades en la taula corresponent i es mostra un missatge informant que la tasca s'ha realitzat.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$posar = db_insert('cinturo')
->fields(array(
'nif' => $record['nif'],
'color' => $form_values['values']['cintcol'],
'dcin' => $form_values['values']['cindia']['day'],
'mcin' => $form_values['values']['cindia']['month'],
'acin' => $form_values['values']['cindia']['year'],
))
->execute();
}
drupal_set_message(t('Dades Publicades correctament.'));
}
```

4.2.2.2 Actualitzar dades federació

· Aquest element serveix per a actualitzar la data de caducitat del carnet de l'associació de Taekwondo d'un client, sense perdre les anteriors. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi.

```
function adm_federa_form_submit($form_id, $form_values) {
```

· En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

```
if ($form_values['values']['passwd']!= 'antilistas'){
return drupal_set_message(t('Segur? jo crec que el password no es el que toca...'));
}
```

- Mostra un missatge amb les dades del formulari.

```
drupal_set_message(t('Nom: %string.', array('%string' => $form_values['values']['nom'])));
```

```
drupal_set_message(t('Primer cognom: %string.', array('%string' =>
$form_values['values']['cognom1'])));
```

```
drupal_set_message(t('Segon cognom: %string.', array('%string' =>
$form_values['values']['cognom2'])));
```

```
drupal_set_message(t('Número de federat: %string.', array('%string' =>
$form_values['values']['federat'])));
```

```
drupal_set_message(t('Caducitat federació: %string.', array('%string' =>
$form_values['values']['tkdia']['day'].'/'.$form_values['values']['tkdia']['month'].'/'.$form_value
s['values']['tkdia']['year'])));
```

- Comprova si el client existeix en la taula client i taekwondo.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');
$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
    ->condition('nom', $form_values['values']['nom'], '=')
    ->condition('cognom1', $form_values['values']['cognom1'], '=')
    ->condition('cognom2', $form_values['values']['cognom2'], '=');
```

```
$result = $query->execute();
```

```
$record = $result->fetchAssoc();
```

```
$query2 = db_select('taekwondo');
```

```
$query2->fields('taekwondo', array('nif', 'federacio', 'dfed', 'mfed',
'afed'))
    ->condition('nif', $record['nif'], '=');
```

```
$result2 = $query2->execute();
```

```
$record2 = $result2->fetchAssoc();
```

```
}
```

- En cas de no existir en client, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```

if ($record['nif']==NULL){

drupal_set_message(t('Aquest usuari no existeix en la base de dades de clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {

        $query0 = db_select('client');

        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))

            ->condition('cognom1', $form_values['values']['cognom1'],'=');


        $result0 = $query0->execute();

        $record0 = $result0->fetchAssoc();

        if ($record0['nif']!=NULL){

            drupal_set_message(t('Possibles clients amb aquest primer cognom en color verd.'),
            'error');

            drupal_set_message(t('Nom: %string.', array('%string' => $record0['nom'].
            '$record0['cognom1']. '$record0['cognom2']')));

                while($record0 = $result0->fetchAssoc()){

                    drupal_set_message(t('Nom: %string.', array('%string' =>
                    $record0['nom']. '$record0['cognom1']. '$record0['cognom2']')));

                }

            }

            return drupal_set_message(t('Torni a realitzar la consulta.'), 'error');

        }

    }

return drupal_set_message(t('Cap usuari amb aquest primer cognom.'), 'error');

}

```

· En cas de no existir en taekwondo, s'informa de que aquest usuari no té dades de taekwondo i es para.

```

if (!$record2['nif']){

return t('Usuari no existeix en la base de dades de Taekwondo.');
```

```

}
drupal_set_message(t('nif: %string.', array('%string' => $record['nif'])));

· En cas de tenir privilegis d'administrador, introdueix les noves dades en
la taula de taekwondo i mostra un missatge conforme la tasca s'ha
completat.

if (in_array('administrator', $GLOBALS['user']->roles)) {
$posar = db_insert('taekwondo')
  ->fields(array(
    'nif' => $record['nif'],
    'federacio' => $form_values['values']['federat'],
    'dfed' => $form_values['values']['tkdia']['day'],
    'mfed' => $form_values['values']['tkdia']['month'],
    'afed' => $form_values['values']['tkdia']['year'],

  ))
  ->execute();
}
drupal_set_message(t('Dades Publicades correctament.'));
}

```

4.2.2.2.3 Consultar dades bancàries

· Aquest element serveix per consultar l'historial de pagaments associat a un client determinat. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi.

```
function adm_visubanc_form_submit($form_id, $form_values) {
```

· En cas de que la contrasenya del formulari no sigui la correcta mostra missatge d'error i es para.

```

if ($form_values['values']['passwd']!= 'antilistos'){

return drupal_set_message(t('Segur? jo crec que el password no es el que toca...'));

}

```

· Agafa el nom i cognoms del client que s'ha introduït al formulari i són

passats per paràmetres a la funció *adm_resubanc()* associada a “*resultatbanca*”.

```
drupal_goto("resultatbanca", array(

    'query' => array('nom'=>$form_values['values']['nom'],
    'cognom1'=>$form_values['values']['cognom1'],
    'cognom2'=>$form_values['values']['cognom2']),

    ));

}
```

```
function adm_resubanc() {
```

- Si l'usuari té permisos d'administrador, consulta si existeix en la taula de clients.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {

$query = db_select('client');

$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif', 'compaga'))

    ->condition('nom', $_GET['nom'], '=')

    ->condition('cognom1', $_GET['cognom1'], '=')

    ->condition('cognom2', $_GET['cognom2'], '=')

$result = $query->execute();

$record = $result->fetchAssoc();

}
```

- En cas de no existir en client, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```
if ($record['nif']==NULL){

drupal_set_message(t('Aquest usuari no existeix en la base de dades de clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {

        $query0 = db_select('client');
```



```

$query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))

->condition('cognom1', $form_values['values']['cognom1'],'=');

$result0 = $query0->execute();

$record0 = $result0->fetchAssoc();

if ($record0['nif']!=NULL){

    drupal_set_message(t('Possibles clients amb aquest primer cognom en color verd.'),
'error');

    drupal_set_message(t('Nom: %string.', array('%string' => $record0['nom'].
'.$record0['cognom1']. '$record0['cognom2']')));

    while($record0 = $result0->fetchAssoc()){

        drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. '$record0['cognom1']. '$record0['cognom2']')));

    }

}

return drupal_set_message(t('Torni a realitzar la consulta.'), 'error');

}

return drupal_set_message(t('Cap usuari amb aquest primer cognom.'), 'error');

}

```

· Agafa les variables de mètode de pagament i inicialitza l'*string output*, on s'emmagatzema el codi *html* que retorna la funció per crear la pàgina.

```

$pagina = aes_decrypt($record['compaga']);

$output = '<p>'.$_GET['nom'].' '.$_GET['cognom1'].' '.$_GET['cognom2'].'</p>';

$output .= '<p>'.$record['nif'].'</p>';

$output .= '<table><th>Import</th><th>Data</th>';

```

· Es consulta la informació dels pagaments i s'inclouen en una taula, en ordre de més nou a més antic.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {

```

```

$query2 = db_select('banc');

$query2->fields('banc', array('nif', 'import', 'dpag', 'mpag', 'apag'))

    ->condition('nif', $record['nif'],'=')

    ->orderBy('apag', 'DESC')

    ->orderBy('mpag', 'DESC');

$result2 = $query2->execute();

while ($record2 = $result2->fetchAssoc()){

    $output .= '<tr>

    <td>'.$record2['import'].' euros</td>

    <td>'.$record2['dpag'].'/'.$record2['mpag'].'/'.$record2['apag'].'</td></tr>';

}

    $output .= '</table>';

}

· Retorna l'string per mostrar en pantalla la pàgina html.

    return $output;

}

```

4.2.3 Gamifica:

El procés de gamificar les activitats és la gran novetat d'aquest projecte. Per tal d'entendre les necessitats de la tasca, algunes idees bàsiques s'han extret de llibres (citats a la bibliografia) respecte l'aplicació d'aquesta mentalitat en un negoci.

La idea principal entorn a la qual gira aquest treball és que els clients actualment ja realitzen aquestes tasques. No es farà que facin res de nou, únicament s'interpretaran els resultats dels seu treball, premiant-los pel seu esforç i per tant fomentant la pràctica i la perseverança. Donar un valor afegit a la motivació que ja els porta a assistir al centre.

Quina és la seva motivació? Millorar o mantenir la seva forma física.

Quines activitats realitzen per a assolir els seus objectius? Exercicis físics (Aixecament de pesos, córrer amb la bicicleta estàtica...). Ells mateixos solen mesurar els seus exercicis i els posen en comú amb els seus amics? Si. Doncs es l'hora de passar al següent nivell. Que el mateix centre promoció la competició entre els seus clients, donant un punt extra tant a la motivació com a la diversió de la tasca en si mateix.

Com a prova pilot d'aquesta funcionalitat, el mòdul següent gestiona campionats d'aixecament de pesos, repetició d'abdominals, distància en bicicleta estàtica i puntuació en partits d'*squash*, mostrant els resultats en una sèrie de *Leaderboards* públiques a tots els usuaris.

4.2.3.1 Info:

Aquest fitxer és el que determina els noms que rebrà el mòdul dins l'assistent de mòduls de *Drupal*. Inclou la descripció, el paquet dins del qual s'englobarà el mòdul (tots els creats per l'autor d'aquest projecte s'engloben dins del paquet "*Drupal 7 Virgili.org*"), i els fitxers que necessita el mòdul per a funcionar. En el fitxer *.install* hi ha les dades d'inicialització de les taules de la base de dades que requereix el mòdul, i en el fitxer *.module* les implementacions del codi que realitza les diferents tasques. Aquest mòdul és dependent dels dos mòduls administratius, ja que s'utilitzen per introduir algunes de les dades que es visualitzen des del mòdul actual.

```
name = Modul de gamificar.  
description = Gamifica el teu portal esportiu.  
dependencies[] = adm_gest  
dependencies[] = adm_gest_add  
package = Drupal 7 Virgili.org  
core = 7.x  
files[] = gamifica.install  
files[] = gamifica.module
```

4.2.3.2 Install:

Aquest fitxer inicialitza les taules dins de la base de dades que

utilitzarà el mòdul per emmagatzemar dades. *Drupal 7* únicament utilitza l'anomenat *hook_schema()* al activar el mòdul i al desinstalar-lo (esborrant el contingut de totes les taules).

```
/**  
 * Implements hook_schema().  
 */
```

· Les taules d'aquest mòdul serveixen per guardar les dades de les diferents competicions en les que poden participar els clients.

```
function gamifica_schema() {
```

· cabdominals emmagatzema les dades de campionat d'abdominals.

```
  $schema['cabdominals'] = array(  
    'description' => 'Taula base per campionat abdominals.',  
    'fields' => array(  
      'num' => array(  
        'description' => 'Numero de la entrada.',  
        'type' => 'serial',  
        'unsigned' => TRUE,  
        'not null' => TRUE,  
      ),  
      'nif' => array(  
        'description' => 'document identitat.',  
        'type' => 'varchar',  
        'length' => 10,  
        'not null' => TRUE,  
        'default' => 0,  
      ),  
      'contador' => array(  
        'description' => 'Contador de exercicis.',  
        'type' => 'int',  
        'size' => 'normal',  
        'not null' => TRUE,  
        'default' => 0,  
      ),  
    ),  
  );
```

```
'dia' => array(
    'description' => 'Dia de publicacio del resultat.',
    'type' => 'int',
    'size' => 'normal',
    'not null' => TRUE,
),
'mes' => array(
    'description' => 'Mes de publicacio del resultat.',
    'type' => 'int',
    'size' => 'normal',
    'not null' => TRUE,
),
'any' => array(
    'description' => 'Any de publicacio del resultat.',
    'type' => 'int',
    'size' => 'normal',
    'not null' => TRUE,
),
),
'primary key' => array('num'),
);
```

- cpeses emmagatzema les dades de campionat de peses.

```
$schema['cpeses'] = array(
    'description' => 'Taula base per campionat peses.',
    'fields' => array(
        'num' => array(
            'description' => 'Numero de la entrada.',
            'type' => 'serial',
            'unsigned' => TRUE,
            'not null' => TRUE,
        ),
        'nif' => array(
            'description' => 'document identitat.',
            'type' => 'varchar',
```

```
'length' => 10,
'not null' => TRUE,
'default' => 0,
),
'pes' => array(
'description' => 'Contador de pes.',
'type' => 'int',
'size' => 'normal',
'not null' => TRUE,
'default' => 0,
),
'contador' => array(
'description' => 'Contador de exercicis.',
'type' => 'int',
'size' => 'normal',
'not null' => TRUE,
'default' => 0,
),
'dia' => array(
'description' => 'Dia de publicacio del resultat.',
'type' => 'int',
'size' => 'normal',
'not null' => TRUE,
),
'mes' => array(
'description' => 'Mes de publicacio del resultat.',
'type' => 'int',
'size' => 'normal',
'not null' => TRUE,
),
'any' => array(
'description' => 'Any de publicacio del resultat.',
'type' => 'int',
'size' => 'normal',
```

```
        'not null' => TRUE,  
    ),  
),  
    'primary key' => array('num'),  
);
```

· cestatica emmagatzema les dades de campionat de bicicleta estàtica.

```
$schema['cestatica'] = array(  
    'description' => 'Taula base per campionat bicicleta estatica.',  
    'fields' => array(  
        'num' => array(  
            'description' => 'Numero de la entrada.',  
            'type' => 'serial',  
            'unsigned' => TRUE,  
            'not null' => TRUE,  
        ),  
        'nif' => array(  
            'description' => 'document identitat.',  
            'type' => 'varchar',  
            'length' => 10,  
            'not null' => TRUE,  
            'default' => 0,  
        ),  
        'contador' => array(  
            'description' => 'Contador de Km.',  
            'type' => 'numeric',  
            'size' => 'normal',  
            'not null' => TRUE,  
            'default' => 0,  
            'precision' => 5,  
            'scale' => 2,  
        ),  
        'dia' => array(  
            'description' => 'Dia de publicacio del resultat.',  
            'type' => 'int',
```

```
        'size' => 'normal',
        'not null' => TRUE,
    ),
    'mes' => array(
        'description' => 'Mes de publicacio del resultat.',
        'type' => 'int',
        'size' => 'normal',
        'not null' => TRUE,
    ),
    'any' => array(
        'description' => 'Any de publicacio del resultat.',
        'type' => 'int',
        'size' => 'normal',
        'not null' => TRUE,
    ),
),
'primary key' => array('num'),
);
```

- csquash emmagatzema els punts dels usuaris que han participat en partits d'squash.

```
$schema['csquash'] = array(
    'description' => 'Taula base per campionat squash.',
    'fields' => array(
        'num' => array(
            'description' => 'Numero de la entrada.',
            'type' => 'serial',
            'unsigned' => TRUE,
            'not null' => TRUE,
        ),
        'nif' => array(
            'description' => 'document identitat jugador.',
            'type' => 'varchar',
            'length' => 10,
            'not null' => TRUE,
```



```
'default' => 0,
),
'punt' => array(
  'description' => 'Puntuacio.',
  'type' => 'int',
  'size' => 'normal',
  'not null' => TRUE,
  'default' => 0,
),
'dia' => array(
  'description' => 'Dia de publicacio del partit.',
  'type' => 'int',
  'size' => 'normal',
  'not null' => TRUE,
),
'mes' => array(
  'description' => 'Mes de publicacio del partit.',
  'type' => 'int',
  'size' => 'normal',
  'not null' => TRUE,
),
'any' => array(
  'description' => 'Any de publicacio del partit.',
  'type' => 'int',
  'size' => 'normal',
  'not null' => TRUE,
),
),
'primary key' => array('num'),
);
```

· csquashmatch relaciona els enfrontaments dels usuaris en una data concreta.

```
$schema['csquashmatch'] = array(
  'description' => 'Taula base per campionat squash.',
```

```
'fields' => array(
  'num' => array(
    'description' => 'Numero de la entrada.',
    'type' => 'serial',
    'unsigned' => TRUE,
    'not null' => TRUE,
  ),
  'nif1' => array(
    'description' => 'document identitat jugador 1.',
    'type' => 'varchar',
    'length' => 10,
    'not null' => TRUE,
    'default' => 0,
  ),
  'nif2' => array(
    'description' => 'document identitat jugador 2.',
    'type' => 'varchar',
    'length' => 10,
    'not null' => TRUE,
    'default' => 0,
  ),
  'dia' => array(
    'description' => 'Dia de publicacio del partit.',
    'type' => 'int',
    'size' => 'normal',
    'not null' => TRUE,
  ),
  'mes' => array(
    'description' => 'Mes de publicacio del partit.',
    'type' => 'int',
    'size' => 'normal',
    'not null' => TRUE,
  ),
  'any' => array(
```

```

        'description' => 'Any de publicacio del partit.',
        'type' => 'int',
        'size' => 'normal',
        'not null' => TRUE,
    ),
),
'primary key' => array('num'),
);

return $schema;
}

```

4.2.3.3 Module:

En aquest fitxer es pot trobar la implementació de les pàgines d'introducció i visualització de dades de campionats. El fitxer comença amb un menú on podem veure les diferents opcions (*[nom_del_mòdul]_menu()*) i on redirecciona cadascuna (*\$items['url']*).

```

function gamifica_menu() {
    $items = array();

    $items['abdominalsintro'] = array(
        'title' => t('Nova entrada'),
        'page callback' => 'camp_abdominals_intro',
        'access arguments' => array('access content'),
        'description' => t('Afegir entrada: campionat abdominals'),
        'type' => MENU_CALLBACK,
    );

    $items['pesesintro'] = array(
        'title' => t('Nova entrada'),
        'page callback' => 'peses_intro',
        'access arguments' => array('access content'),
        'description' => t('Afegir entrada: campionat peses'),
        'type' => MENU_CALLBACK,
    );

    $items['staticintro'] = array(

```

```
'title' => t('Nova entrada'),
'page callback' => 'static_intro',
'access arguments' => array('access content'),
'description' => t('Afegir entrada: campionat estàtica'),
'type' => MENU_CALLBACK,
);
$items['squashintro'] = array(
'title' => t('Nova entrada'),
'page callback' => 'squash_intro',
'access arguments' => array('access content'),
'description' => t('Afegir entrada: partida squash'),
'type' => MENU_CALLBACK,
);
$items['abdominalssee'] = array(
'title' => t('Resultats Campionat Abdominals'),
'page callback' => 'camp_abdominals_see',
'access arguments' => array('access content'),
'description' => t('Resultats campionat abdominals'),
'type' => MENU_CALLBACK,
);
$items['pesessee'] = array(
'title' => t('Resultats Campionat Peses'),
'page callback' => 'peses_see',
'access arguments' => array('access content'),
'description' => t('Resultats campionat peses'),
'type' => MENU_CALLBACK,
);
$items['staticsee'] = array(
'title' => t('Resultats Campionat Estàtica'),
'page callback' => 'static_see',
'access arguments' => array('access content'),
'description' => t('Resultats campionat estàtica'),
'type' => MENU_CALLBACK,
);
```

```

$items['squashsee'] = array(
  'title' => t('Resultats Partides Squash'),
  'page callback' => 'squash_see',
  'access arguments' => array('access content'),
  'description' => t('Resultats partides squash'),
  'type' => MENU_CALLBACK,
);
return $items;
}

```

4.2.3.3.1 Afegir entrada: campionat abdominals

· El primer dels menús porta a la introducció de noves dades de campionat d'abdominals en el sistema. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function camp_abdominals_form_submit($form_id, $form_values) {
```

· Comprova si l'usuari existeix en la taula client.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');
$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
  ->condition('nom', $form_values['values']['nom'],'=')
  ->condition('cognom1', $form_values['values']['cognom1'],'=')
  ->condition('cognom2', $form_values['values']['cognom2'],'=');

```

```

$result = $query->execute();
$record = $result->fetchAssoc();
}

```

· En cas de no existir en client, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```

if ($record['nif']==NULL){
  drupal_set_message(t('Aquest usuari no existeix en la base de dades de clients.'), 'error');

```

```

  if (in_array('administrator', $GLOBALS['user']->roles)) {

```

```

$query0 = db_select('client');
$query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
->condition('cognom1', $form_values['values']['cognom1'],'=');

$result0 = $query0->execute();
$record0 = $result0->fetchAssoc();

if ($record0['nif']!=NULL){
    drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');

    drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2'])));

    while($record0 = $result0->fetchAssoc()){
        drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2'])));
    }
}

return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');

}

return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
'error');
}

```

- Mostra un missatge amb el contingut del formulari.

```

drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom']. ' '. $form_values['values']['cognom1']. '
'. $form_values['values']['cognom2'])));

drupal_set_message(t('Exercicis %string.', array('%string' =>
$form_values['values']['contador'])));

drupal_set_message(t('Dia %string.', array('%string' =>
$form_values['values']['dia']['day'])));

drupal_set_message(t('Mes %string.', array('%string' =>
$form_values['values']['dia']['month'])));

drupal_set_message(t('Any %string.', array('%string' =>
$form_values['values']['dia']['year'])));

```

- Si l'usuari té permisos d'administrador, introdueix les dades del campionat en la base de dades i mostra un missatge conforme s'ha dut a terme la tasca.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {

```

```

$posar = db_insert('cabdominals')
->fields(array(
    'nif' => $record['nif'],
    'contador' => $form_values['values']['contador'],
    'dia' => $form_values['values']['dia']['day'],
    'mes' => $form_values['values']['dia']['month'],
    'any' => $form_values['values']['dia']['year'],
))
->execute();
}
drupal_set_message(t('Dades Publicades correctament.'));
}

```

4.2.3.3.2 Afegir entrada: campionat peses

· Aquest menú porta a la introducció de noves dades de campionat de peses en el sistema. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```

function peses_form_submit($form_id, $form_values) {
    · Comprova si l'usuari existeix en la taula clients.

    //Comprobar si existeix l'usuari

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query = db_select('client');
        $query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('nom', $form_values['values']['nom'],'=')
            ->condition('cognom1', $form_values['values']['cognom1'],'=')
            ->condition('cognom2', $form_values['values']['cognom2'],'=');
        $result = $query->execute();
        $record = $result->fetchAssoc();
    }
}

```

· En cas de no existir en client, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a

possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```
if ($record['nif']==NULL){
    drupal_set_message(t('Aquest usuari no existeix en la base de dades de
clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query0 = db_select('client');
        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['cognom1'], '=');
        $result0 = $query0->execute();
        $record0 = $result0->fetchAssoc();
        if ($record0['nif']!=NULL){
            drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');
            drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2'])));
            while($record0 = $result0->fetchAssoc()){
                drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2'])));
            }
        }
        return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');
    }
    return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
'error');
}
```

· Mostra les dades del formulari per pantalla.

```
drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom']. ' '. $form_values['values']['cognom1']. '
'. $form_values['values']['cognom2'])));

drupal_set_message(t('Pes: %string.', array('%string' =>
$form_values['values']['pes'])));

drupal_set_message(t('Exercicis: %string.', array('%string' =>
$form_values['values']['contador'])));

drupal_set_message(t('Dia %string.', array('%string' =>
$form_values['values']['dia']['day'])));

drupal_set_message(t('Mes %string.', array('%string' =>
```



```
$form_values['values']['dia']['month'])));
drupal_set_message(t('Any %string.', array('%string' =>
$form_values['values']['dia']['year'])));
```

- En cas de que l'usuari tingui permisos d'administrador, introdueix les dades del campionat en la taula de peses i mostra un missatge informant de la realització de la tasca.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$posar = db_insert('cpeses')
->fields(array(
'nif' => $record['nif'],
'pes' => $form_values['values']['pes'],
'contador' => $form_values['values']['contador'],
'dia' => $form_values['values']['dia']['day'],
'mes' => $form_values['values']['dia']['month'],
'any' => $form_values['values']['dia']['year'],
))
->execute();
}
drupal_set_message(t('Dades Publicades correctament.'));
}
```

4.2.3.3.3 Afegir entrada: campionat bicicleta estàtica

- Aquest menú porta a la introducció de noves dades de campionat de bicicleta estàtica en el sistema. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function static_form_submit($form_id, $form_values) {

//Comprobar si existeix l'usuari

· Comprova si l'usuari existeix en la taula client.

if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');
$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
```

```

->condition('nom', $form_values['values']['nom'],'=')
->condition('cognom1', $form_values['values']['cognom1'],'=')
->condition('cognom2', $form_values['values']['cognom2'],'=');
$result = $query->execute();
$record = $result->fetchAssoc();
}

· En cas de no existir en client, es realitza una cerca del primer cognom. Si
troba clients amb aquest primer cognom, els mostra per pantalla com a
possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

if ($record['nif']==NULL){
    drupal_set_message(t('Aquest usuari no existeix en la base de dades de
clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query0 = db_select('client');
        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['cognom1'],'=');
        $result0 = $query0->execute();
        $record0 = $result0->fetchAssoc();
        if ($record0['nif']!=NULL){
            drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');
            drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom'].', '. $record0['cognom1'].', '. $record0['cognom2'])));
            while($record0 = $result0->fetchAssoc()){
                drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom'].', '. $record0['cognom1'].', '. $record0['cognom2'])));
            }
        }
        return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');
    }

    return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
'error');
}

```

- Mostra les dades del formulari en pantalla.

```
drupal_set_message(t('Nom: %string.', array('%string' =>
$form_values['values']['nom'].' ': $form_values['values']['cognom1'].'
' . $form_values['values']['cognom2']))););

drupal_set_message(t('Km: %string.', array('%string' =>
$form_values['values']['contador']))););

drupal_set_message(t('Dia %string.', array('%string' =>
$form_values['values']['dia']['day']))););

drupal_set_message(t('Mes %string.', array('%string' =>
$form_values['values']['dia']['month']))););

drupal_set_message(t('Any %string.', array('%string' =>
$form_values['values']['dia']['year']))););
```

- En cas de tenir privilegis d'administrador, introdueix les dades del campionat en la taula de bicicleta estàtica i mostra un missatge informant de que la tasca ha estat realitzada.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$posar = db_insert('cestatica')
->fields(array(
'nif' => $record['nif'],
'contador' => $form_values['values']['contador'],
'dia' => $form_values['values']['dia']['day'],
'mes' => $form_values['values']['dia']['month'],
'any' => $form_values['values']['dia']['year'],
))
->execute();
}
drupal_set_message(t('Dades Publicades correctament.'));
}
```

4.2.3.3.4 Afegir entrada: campionat squash

- Aquest menú porta a la introducció de noves dades de campionat de *squash* en el sistema. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function squash_form_submit($form_id, $form_values) {
```

```
//Comprobar si existeix el jugador 1
```

- Comprova si el primer usuari existeix en la taula client.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');
$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
    ->condition('nom', $form_values['values']['p1n'], '=')
    ->condition('cognom1', $form_values['values']['p1c1'], '=')
    ->condition('cognom2', $form_values['values']['p1c2'], '=');

$result = $query->execute();
$record = $result->fetchAssoc();
}
```

- En cas de no existir en client, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```
if ($record['nif']==NULL){
    drupal_set_message(t('El Jugador 1 no existeix en la base de dades de
clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query0 = db_select('client');
        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['p1c1'], '=');
        $result0 = $query0->execute();
        $record0 = $result0->fetchAssoc();
        if ($record0['nif']!=NULL){
            drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');

            drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2']))) );
            while($record0 = $result0->fetchAssoc()){
                drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2']))) );
            }
        }
    }
}
```

```

    }

    return drupal_set_message(t('Torni a realitzar la consulta.'),
    'error');

    }

    return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
    'error');

}

```

//Comprobar si existeix el jugador 2

- Comprova si el segon usuari existeix en la taula client.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {
$query2 = db_select('client');
$query2->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
    ->condition('nom', $form_values['values']['p2n'], '=')
    ->condition('cognom1', $form_values['values']['p2c1'], '=')
    ->condition('cognom2', $form_values['values']['p2c2'], '=');

```

```

$result2 = $query2->execute();
$record2 = $result2->fetchAssoc();
}

```

- En cas de no existir en client, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```

if ($record2['nif']==NULL){
    drupal_set_message(t('El Jugador 2 no existeix en la base de dades de
clients.'), 'error');

```

```

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query0 = db_select('client');
        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['p2c1'], '=');

        $result0 = $query0->execute();
    }

```

```

$record0 = $result0->fetchAssoc();

if ($record0['nif']!=NULL){

    drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');

    drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2'])));

    while($record0 = $result0->fetchAssoc()){

        drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom']. ' '. $record0['cognom1']. ' '. $record0['cognom2'])));

    }

}

return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');

}

return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
'error');

}

```

- Mostra en pantalla un missatge amb les dades del formulari.

```

drupal_set_message(t('Jugador 1: %string.', array('%string' =>
$form_values['values']['p1n']. ' '. $form_values['values']['p1c1']. '
'. $form_values['values']['p1c2'])));

drupal_set_message(t('Puntuació: %string.', array('%string' =>
$form_values['values']['jocsp1'])));

drupal_set_message(t('Jugador 2: %string.', array('%string' =>
$form_values['values']['p2n']. ' '. $form_values['values']['p2c1']. '
'. $form_values['values']['p2c2'])));

drupal_set_message(t('Puntuació: %string.', array('%string' =>
$form_values['values']['jocsp2'])));

drupal_set_message(t('Dia %string.', array('%string' =>
$form_values['values']['dia']['day'])));

drupal_set_message(t('Mes %string.', array('%string' =>
$form_values['values']['dia']['month'])));

drupal_set_message(t('Any %string.', array('%string' =>
$form_values['values']['dia']['year'])));

```

- En cas de tenir privilegis d'administrador, s'inserta les dades del partit en la taula d'historial d'enfrontaments.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {

$posar0 = db_insert('csquashmatch')

->fields(array(

```

```

'nif1' => $record['nif'],
'nif2' => $record2['nif'],
'dia' => $form_values['values']['dia']['day'],
'mes' => $form_values['values']['dia']['month'],
'any' => $form_values['values']['dia']['year'],
))
->execute();

```

- En cas de tenir privilegis d'administrador, s'inserta les dades de puntuació de cada usuari en la taula d'historial de punts i s'informa de que la tasca ha estat completada.

```

$posar = db_insert('csquash')
->fields(array(
  'nif' => $record['nif'],
  'punt' => $form_values['values']['jocsp1'],
  'dia' => $form_values['values']['dia']['day'],
  'mes' => $form_values['values']['dia']['month'],
  'any' => $form_values['values']['dia']['year'],
))
->execute();

$posar = db_insert('csquash')
->fields(array(
  'nif' => $record2['nif'],
  'punt' => $form_values['values']['jocsp2'],
  'dia' => $form_values['values']['dia']['day'],
  'mes' => $form_values['values']['dia']['month'],
  'any' => $form_values['values']['dia']['year'],
))
->execute();
}
drupal_set_message(t('Dades Publicades correctament.'));
}

```

4.2.3.3.5 Resultats campionat abdominals

· Aquest menú porta a la visualització de les dades de campionat d'abdominals en el sistema. Retorna una pàgina en *html* mitjançant el següent codi:

```
function camp_abdominals_see() {
```

· Inicialitza l'array que retornarà el codi a incloure dins de la variable de la constructora *content*.

```
$output = '';
```

· Extreu totes les dades de la taula d'abdominals i les ordena pel comptador d'exercicis, de gran a petit.

```
    $query = db_select('cabdominals');
    $query->fields('cabdominals', array('nif', 'contador', 'dia', 'mes',
    'any'))
        ->orderBy('contador', 'DESC'); //ORDER BY contador
    $result = $query->execute();
```

· Inicialitza el contador de posició.

```
$cont = 0;
```

```
$output .= '<table>';
```

· Mentre no s'hagi acabat el llistat amb el valor de la consulta.

```
while ($record = $result->fetchAssoc()){
```

· Posició del jugador s'incrementa.

```
    $cont = $cont+1;
```

· S'obté el seu nom de la taula client.

```
        //obtenir el nom del usuari
        $query2 = db_select('client');
        $query2->fields('client', array('nif', 'nom', 'cognom1', 'cognom2'))
            ->condition('nif', $record['nif'],'=');

        $result2 = $query2->execute();
        $record2 = $result2->fetchAssoc();
```


- Els tres primers tenen podi amb medalla creada per l'ocasió.

```

        if($cont <= 3){
            $output .= '
                <td><br>
                ' . $record2['nom'] . ' ' . $record2['cognom1'] . '
' . $record2['cognom2'] . '<br>
                ' . $cont . 'º: ' . $record['contador'] . ' Repeticions<br>
                ' . $record['dia'] . '/' . $record['mes'] . '/' . $record['any'] . '</td>';
            }

```

- A partir del quart, es crea una taula on es posa posició, nom, exercicis i data d'assoliment.

```

        if($cont == 4){
            $output .=
'</table><table><th>Pos.</th><th>Usuari</th><th>Repeticions</th><th>data</th>
';

            $output .= '<tr><td>' . $cont . '</td>
                <td>' . $record2['nom'] . ' ' . $record2['cognom1'] . '
' . $record2['cognom2'] . '</td>
                <td>' . $record['contador'] . '</td>
                <td>' . $record['dia'] . '/' . $record['mes'] . '/' . $record['any'] . '</td></tr>';
            }

            if($cont > 4){
                $output .= '<tr><td>' . $cont . '</td>
                    <td>' . $record2['nom'] . ' ' . $record2['cognom1'] . '
' . $record2['cognom2'] . '</td>
                    <td>' . $record['contador'] . '</td>
                    <td>' . $record['dia'] . '/' . $record['mes'] . '/' . $record['any'] . '</td></tr>';
                }
            }
            $output .= '</table>';

```

- Es retorna el codi *html* per mostrar.

```
        return $output;
    }
}
```

4.2.3.3.6 Resultats campionat peses

- Aquest menú porta a la visualització de les dades de campionat de peses en el sistema. Retorna una pàgina en *html* mitjançant el següent codi:

```
function peses_see() {
```

- Inicialitza l'array que retornarà el codi a incloure dins de la variable de la constructora *content*.

```
$output = '';
```

- Extreu totes les dades de la taula de peses i les ordena pel pes primer, i pel contador d'exercicis després, de més gran a més petit.

```
    $query = db_select('cpeses');
    $query->fields('cpeses', array('nif', 'pes', 'contador', 'dia', 'mes',
    'any'))
        ->orderBy('pes', 'DESC') //ORDER BY contador
        ->orderBy('contador', 'DESC'); //ORDER BY contador
```

```
    $result = $query->execute();
```

- Inicialitza el contador de posició.

```
$cont = 0;
```

```
$output .= '<table>';
```

- Mentre no s'hagi acabat el llistat amb el valor de la consulta.

```
while ($record = $result->fetchAssoc()){
```

- Posició del jugador s'incrementa.

```
    $cont = $cont+1;
```

- S'obté el seu nom de la taula client.

```
        //obtenir el nom del usuari
        $query2 = db_select('client');
```

```
$query2->fields('client', array('nif', 'nom', 'cognom1', 'cognom2'))
->condition('nif', $record['nif'], '=');
```

```
$result2 = $query2->execute();
$record2 = $result2->fetchAssoc();
```

- Els tres primers tenen podi amb medalla creada per l'ocasió.

```
if($cont <= 3){
    $output .= '
        <td><br>
        '.$record2['nom'].' '.$record2['cognom1'].'
        '.$record2['cognom2'].'<br>
        '.$record['pes'].' Kg. '.$record['contador'].' Repeticions<br>
        '.$record['dia'].'/'.$record['mes'].'/'.$record['any'].'</td>';
    }
}
```

- A partir del quart, es crea una taula on es posa posició, nom, exercicis i data d'assoliment.

```
if($cont == 4){
    $output .=
'</table><table><th>Pos.</th><th>Usuari</th><th>Kg.</th><th>Repeticions</th><
th>data</th>';

    $output .= '<tr><td>'.$cont.'</td>
        <td>'.$record2['nom'].' '.$record2['cognom1'].'
        '.$record2['cognom2'].'</td>
        <td>'.$record['pes'].'</td>
        <td>'.$record['contador'].'</td>
        <td>'.$record['dia'].'/'.$record['mes'].'/'.$record['any'].'</td></tr>';
    }

    if($cont > 4){

        $output .= '<tr><td>'.$cont.'</td>
            <td>'.$record2['nom'].' '.$record2['cognom1'].'
            '.$record2['cognom2'].'</td>
```

```

        <td>'.'.$record['pes'].'</td>
        <td>'.'.$record['contador'].'</td>

<td>'.'.$record['dia'].'/'.$record['mes'].'/'.$record['any'].'</td></tr>';
    }
}
$output .= '</table>';

    return $output;
}

```

4.2.3.3.7 Resultats campionat bicicleta estàtica

· Aquest menú porta a la visualització de les dades de campionat de bicicleta estàtica en el sistema. Retorna una pàgina en *html* mitjançant el següent codi:

```
function static_see() {
```

· Inicialitza l'array que retornarà el codi a incloure dins de la variable de la constructora *content*.

```
$output = '';
```

· Extreu totes les dades de la taula de bicicleta estàtica i les agrupa pel NIF o NIE, entre els quals suma les distàncies si pertanyen al mateix usuari, i les ordena pel comptador de distància resultant, de més gran a petit.

```

$query = db_select('cestatica');
$query->fields('cestatica', array('nif', 'contador'));
$query->groupBy('nif');
$query->addExpression('SUM(contador)', 'contador');
$query->orderBy('contador', 'DESC');

```

```
$result = $query->execute();
```

· Inicialitza el comptador de posició.

```
$cont = 0;
```

```
$output .= '<table>';
```

- Mentre no s'hagi acabat el llistat amb el valor de la consulta.

```
while ($record = $result->fetchAssoc()){
```

- Posició del jugador s'incrementa.

```
    $cont = $cont+1;
```

- S'obté el seu nom de la taula client.

```
    //obtenir el nom del usuari
```

```
    $query2 = db_select('client');
```

```
    $query2->fields('client', array('nif', 'nom', 'cognom1', 'cognom2'))
```

```
        ->condition('nif', $record['nif'],'=');
```

```
    $result2 = $query2->execute();
```

```
    $record2 = $result2->fetchAssoc();
```

- Els tres primers tenen podi amb medalla creada per l'ocasió.

```
    if($cont <= 3){
```

```
        $output .= '
```

```
        <td><br>
```

```
        '.$record2['nom']. ' '.$record2['cognom1']. '
'.$record2['cognom2']. '<br>
```

```
        '.$cont.'º: '.$record['contador']. ' Km.</td>';
```

```
    }
```

- A partir del quart, es crea una taula on es posa posició, nom i quilòmetres.

```
    if($cont == 4){
```

```
        $output .=
'</table><table><th>Pos.</th><th>Usuari</th><th>Km.</th>';
```

```
        $output .= '<tr><td>'.$cont.'</td>
```

```
        <td>'.$record2['nom']. ' '.$record2['cognom1']. '
'.$record2['cognom2']. '</td>
```

```
        <td>'.$record['contador']. ' Km.</td></tr>';
```

```
    }
```

```

        if($cont > 4){

            $output .= '<tr><td>'.$cont.'</td>

                <td>'.$record2['nom'].' '.$record2['cognom1'].'
            '.$record2['cognom2'].'</td>

                <td>'.$record['contador'].' Km.</td></tr>';

            }

        }

        $output .= '</table>';

    · Es retorna el codi html per mostrar.

        return $output;

    }

```

4.2.3.3.8 Resultats campionat squash

· Aquest menú porta a la visualització de les dades de puntuació dels diferents usuaris que han participat en els partits d'squash. Retorna una pàgina en *html* mitjançant el següent codi:

```
function squash_see() {
```

· Inicialitza l'array que retornarà el codi a incloure dins de la variable de la constructora *content*.

```
$output = '';
```

· Extreu totes les dades dels usuaris de la taula d'usuaris d'squash i els agrupa per jugadors, sumant les seves puntuacions, que són ordenades de gran a petit.

```

$query = db_select('csquash');
$query->fields('csquash', array('nif', 'punt'));
$query->groupBy('nif');
$query->addExpression('SUM(punt)', 'punt');
$query->orderBy('punt', 'DESC'); //ORDER BY contador

```

```
$result = $query->execute();
```

- Inicialitza el contador de posició.

```
$cont = 0;
```

```
$output .= '<table>';
```

- Mentre no s'hagi acabat el llistat amb el valor de la consulta.

```
while ($record = $result->fetchAssoc()){
```

- Posició del jugador s'incrementa.

```
$cont = $cont+1;
```

- S'obté el seu nom de la taula client.

```
//obtenir el nom del usuari
```

```
$query2 = db_select('client');
```

```
$query2->fields('client', array('nif', 'nom', 'cognom1', 'cognom2'))
```

```
->condition('nif', $record['nif'],'=');
```

```
$result2 = $query2->execute();
```

```
$record2 = $result2->fetchAssoc();
```

- Els tres primers tenen podi amb medalla creada per l'ocasió.

```
if($cont <= 3){
```

```
$output .= '
```

```
<td><br>
```

```
'.$record2['nom']. ' '.$record2['cognom1']. '
'.$record2['cognom2']. '<br>
```

```
'.$cont.'º: '.$record['punt']. ' Punts.</td>';
```

```
}
```

- A partir del quart, es crea una taula on es posa posició, nom i exercicis.

```
if($cont == 4){
```

```
$output .=
'</table><table><th>Pos.</th><th>Usuari</th><th>Km.</th>';
```

```
$output .= '<tr><td>'.$cont.'</td>
```

```

        <td>'. $record2['nom']. ' ' $record2['cognom1']. '
        ' $record2['cognom2']. ' </td>

        <td>'. $record['punt']. ' Punts. </td></tr>';

    }

    if($cont > 4){

        $output .= '<tr><td>'. $cont. '</td>

        <td>'. $record2['nom']. ' ' $record2['cognom1']. '
        ' $record2['cognom2']. ' </td>

        <td>'. $record['punt']. ' Punts. </td></tr>';

    }

}

$output .= '</table>';

```

- Es retorna el codi *html* per mostrar.

```

    return $output;
}

```

4.2.4 Mòdul Usuari:

El mòdul d'usuari gestiona el segon apartat de les funcionalitats de gamificació: les fites. Les fites són constatacions de que un usuari ha assolit un objectiu concret. Aquestes poden ser temporals (per exemple donar una medalla per assistència persistent els últims 30 dies, que perdrà en cas de que no pugui continuar) o permanents (una medalla commemorativa del primer dia que un client s'ha caigut de la cinta de córrer, o que ha aixecat més de 80 Kg de pes). Per mantenir l'interès dels usuaris, s'anirà creant noves fites (tot i que en primera instància, únicament pels administradors) les quals poden ser visualitzades per l'usuari en el seu perfil. Com a prova s'han creat una sèrie de vuit fites (amb les seves corresponents medalles) per a tirar endavant aquesta idea, que s'anirà ampliant amb el suport i assistència del centre.

4.2.4.1 Info:

Aquest fitxer és el que determina els noms que rebrà el mòdul dins l'assistent de mòduls de *Drupal*. Inclou la descripció, el paquet dins del qual s'englobarà el mòdul (tots els creats per l'autor d'aquest projecte

s'engloben dins del paquet “*Drupal 7 Virgili.org*”), i els fitxers que necessita el mòdul per a funcionar. En el fitxer *.install* hi ha les dades d'inicialització de les taules de la base de dades que requereix el mòdul, i en el fitxer *.module* les implementacions del codi que realitza les diferents tasques. Aquest mòdul és dependent dels dos mòduls administratius, ja que s'utilitzen per introduir algunes de les dades que es visualitzen des del mòdul actual, i del anterior mòdul de gamificació, pel mateix motiu.

```
name = Modul de Usuari.
description = Gamifica el teu portal esportiu.
dependencies[] = adm_gest
dependencies[] = adm_gest_add
dependencies[] = gamifica
package = Drupal 7 Virgili.org
core = 7.x
files[] = gamificau.install
files[] = gamificau.module
```

4.2.4.2 Install:

Aquest fitxer inicialitza les taules dins de la base de dades que utilitzarà el mòdul per emmagatzemar dades. *Drupal 7* únicament utilitza l'anomenat *hook_schema()* al activar el mòdul i al desinstalar-lo (esborrant el contingut de totes les taules).

```
/**
 * Implements hook_schema().
 */
```

- Les taules d'aquest mòdul serveixen desenvolupar l'apartat de fites, tant la seva creació com la posterior associació amb un usuari concret.

```
function gamificau_schema() {
```

- logro enmagatzema les diferents fites que es poden crear, associant-les a una descripció i una imatge.

```
  $schema['logro'] = array(
    'description' => 'Taula base per a acievements.',
```

```
'fields' => array(
    'num' => array(
        'description' => 'Numero de la entrada.',
        'type' => 'serial',
        'unsigned' => TRUE,
        'not null' => TRUE,
    ),
    'nom' => array(
        'description' => 'Nom del logro.',
        'type' => 'varchar',
        'length' => 20,
        'not null' => TRUE,
        'default' => 0,
    ),
    'descripcio' => array(
        'description' => 'Descripcio del logro.',
        'type' => 'varchar',
        'length' => 256,
        'not null' => TRUE,
        'default' => 0,
    ),
    'noming' => array(
        'description' => 'Nom de la imatge associada.',
        'type' => 'varchar',
        'length' => 16,
        'not null' => TRUE,
        'default' => 0,
    ),
),
'primary key' => array('num'),
);
```

- logrouser emmagatzema les associacions entre fites i els usuaris que les

han aconseguit.

```
$schema['logrouser'] = array(
    'description' => 'Taula user per a acievements.',
    'fields' => array(
        'num' => array(
            'description' => 'Numero de la entrada.',
            'type' => 'serial',
            'unsigned' => TRUE,
            'not null' => TRUE,
        ),
        'nif' => array(
            'description' => 'document identitat.',
            'type' => 'varchar',
            'length' => 10,
            'not null' => TRUE,
            'default' => 0,
        ),
        'numlogro' => array(
            'description' => 'logro assolit.',
            'type' => 'int',
            'size' => 'big',
            'unsigned' => TRUE,
            'not null' => TRUE,
            'default' => 0,
        ),
    ),
    'primary key' => array('num'),
);
return $schema;
}
```

4.2.4.3 Module:

En aquest fitxer es pot trobar la implementació de les pàgines

d'introducció i visualització de dades de campionats. El fitxer comença amb un menú on podem veure les diferents opcions (*[nom_del_mòdul]_menu()*) i on redirecciona cadascuna (*\$items['url']*).

```
function gamificau_menu() {  
    $items = array();  
    $items['logrointro'] = array(  
        'title' => t('Nova fita'),  
        'page callback' => 'logro_intro',  
        'access arguments' => array('access content'),  
        'description' => t('Afegir entrada: Logro'),  
        'type' => MENU_CALLBACK,  
    );  
    $items['userlgintro'] = array(  
        'title' => t('associar fita'),  
        'page callback' => 'userlogro_intro',  
        'access arguments' => array('access content'),  
        'description' => t('Associar usuari: Logro'),  
        'type' => MENU_CALLBACK,  
    );  
    $items['userlogro'] = array(  
        'title' => t('Gaming'),  
        'page callback' => 'userlogro_see',  
        'access arguments' => array('access content'),  
        'description' => t('Resultats usuari'),  
        'type' => MENU_CALLBACK,  
    );  
    return $items;  
}
```

4.2.4.3.1 Nova Fita

· El primer dels menús porta a la creació d'una nova fita en el sistema. Es retorna un formulari amb dades per emplenar i en clicar el botó *Publicar* es posa en marxa el següent codi:

```
function logro_form_submit($form_id, $form_values) {
```

- Comprova si la fita existeix en la taula de fites.

```
//Comprobar si existeix logro
```

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
```

```
$query = db_select('logro');
```

```
$query->fields('logro', array('nom'))
```

```
    ->condition('nom', $form_values['values']['nom'],'=');
```

```
$result = $query->execute();
```

```
$record = $result->fetchAssoc();
```

```
}
```

- En cas d' existir, es mostra un missatge d'error informant de la seva existència i es para.

```
if ($record['nom']!=NULL){
```

```
    return drupal_set_message(t('Aquest logro ja existeix en la base de dades.'), 'error');
```

```
}
```

- Mostra els resultats del formulari per pantalla.

```
drupal_set_message(t('Nom: %string.', array('%string' => $form_values['values']['nom'])));
```

```
drupal_set_message(t('Nom Imatge: %string.', array('%string' => $form_values['values']['nomimg'])));
```

```
drupal_set_message(t('Descripció: %string.', array('%string' => $form_values['values']['comment'])));
```

- Si l'usuari té permisos d'administrador, introdueix la nova fita en la taula corresponent i mostra un missatge conforme s'ha dut a terme la tasca.

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
```

```
$posar = db_insert('logro')
```

```
    ->fields(array(
```

```
        'nom' => $form_values['values']['nom'],
```

```
        'descripcio' => $form_values['values']['comment'],
```

```
        'nomimg' => $form_values['values']['nomimg'],
```

```
    ))
```

```
->execute();
}
drupal_set_message(t('Dades Publicades correctament.'));
}
```

4.2.4.3.2 Associar fita

· Aquest menú serveix per a donar una medalla o nova fita a un usuari. Abans de retornar el formulari, es llegeix de la taula de fites les fites disponibles per a seleccionar:

```
function userlogro_intro() {
```

· Llegeix totes les fites disponibles en la taula de fites:

```
$query = db_select('logro');
$query->fields('logro', array('num', 'nom'));
$resultx = $query->execute();
```

· Es guarden en una variable que es passarà al formulari:

```
while ($recordx = $resultx->fetchAssoc()){
    $pasar[$recordx['num']] = $recordx['nom'];
}
```

· Es retorna un formulari amb dades per emplenar, en el que un dels camps serà un desplegable amb les diferents fites:

```
return drupal_get_form('userlogro_form', $pasar);
}
```

· En clicar el botó *Publicar* es posa en marxa el següent codi:

```
function userlogro_form_submit($form_id, $form_values) {
```

```
//Comprobar si existeix user
```

· Es comprova si existeix l'usuari en la taula de clients:

```
if (in_array('administrator', $GLOBALS['user']->roles)) {
$query = db_select('client');
```

```

$query->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
    ->condition('nom', $form_values['values']['nom'],'=')
    ->condition('cognom1', $form_values['values']['cognom1'],'=')
    ->condition('cognom2', $form_values['values']['cognom2'],'=');

```

```

$result = $query->execute();
$record = $result->fetchAssoc();
}

```

· En cas de no existir en client, es realitza una cerca del primer cognom. Si troba clients amb aquest primer cognom, els mostra per pantalla com a possibles resultats i insta a l'administrador a tornar a realitzar la cerca.

```

if ($record['nif']==NULL){
    drupal_set_message(t('Aquest usuari no existeix en la base de dades de
clients.'), 'error');

    if (in_array('administrator', $GLOBALS['user']->roles)) {
        $query0 = db_select('client');
        $query0->fields('client', array('nom', 'cognom1', 'cognom2', 'nif'))
            ->condition('cognom1', $form_values['values']['cognom1'],'=');

        $result0 = $query0->execute();
        $record0 = $result0->fetchAssoc();
        if ($record0['nif']!=NULL){
            drupal_set_message(t('Possibles clients amb aquest primer cognom en
color verd.'), 'error');

            drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom'].'- '.$record0['cognom1'].'- '.$record0['cognom2'])));
            while($record0 = $result0->fetchAssoc()){
                drupal_set_message(t('Nom: %string.', array('%string' =>
$record0['nom'].'- '.$record0['cognom1'].'- '.$record0['cognom2'])));
            }
        }
        return drupal_set_message(t('Torni a realitzar la consulta.'),
'error');
    }
}

```

```

    }

    return drupal_set_message(t('Cap usuari amb aquest primer cognom.'),
    'error');

}

```

- busca en la base de dades la fita seleccionada.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {
$query2 = db_select('logro');
$query2->fields('logro', array('num', 'nom'))
    ->condition('num', $form_values['values']['nomlogro'], '=');

$result2 = $query2->execute();
$record2 = $result2->fetchAssoc();
}

```

- Mostra un missatge amb les dades de la fita i l'usuari al que s'associa.

```

drupal_set_message(t('Nom: %string.', array('%string' => $record['nom'].
'.'. $record['cognom1']. ' ' . $record['cognom2']))));

drupal_set_message(t('Formu logro: %string.', array('%string' =>
$form_values['values']['nomlogro'])));

drupal_set_message(t('Nom logro: %string.', array('%string' =>
$record2['nom'])));

```

- Si l'usuari té permisos d'administrador, introdueix la nova associació en la taula de fites d'usuari i mostra un missatge conforme s'ha dut a terme la tasca.

```

if (in_array('administrator', $GLOBALS['user']->roles)) {
$posar = db_insert('logrouser')
    ->fields(array(
        'nif' => $record['nif'],
        'numlogro' => $form_values['values']['nomlogro'],
    ))
    ->execute();
}

drupal_set_message(t('Dades Publicades correctament.'));
}

```


4.2.4.3.3 Gaming

· L'opció d'aquest menú únicament permet la visualització de dades pròpies, i s'ha de posar el DNI o NIE en la compta del portal per a poder accedir-hi. La pàgina resultant es crea a partir d'aquest codi:

```
function userlogro_see() {
```

· Obté el DNI o NIE de la compta de l'usuari que està accedint a la funció, en cas d'estar buit informa a l'usuari de que es posi en contacte amb el centre per a que li activin el servei.

```
// inicialitzar usuari, obtenir dni
```

```
$user_fields = user_load($GLOBALS['user']->uid);
```

```
if ($user_fields->field_nif==NULL){
```

```
return drupal_set_message(t('Contacta amb el gimnàs per a que habilitin el compte.'), 'error');
```

```
}
```

· Si l'usuari no té fotografia de l'apartat *gaming* en el seu perfil, se li assigna una per defecte.

```
$nifactual = $user_fields->field_nif['und']['0']['value'];
```

```
if ($user_fields->field_fotografia==NULL){
```

```
$fotoactual='gaming.png';
```

```
$wactual = 200 px;
```

```
$hactual = 200 px;
```

```
}
```

· Si l'usuari té fotografia de l'apartat *gaming* en el seu perfil, s'agafa el nom i les mides.

```
$fotoactual = $user_fields->field_fotografia['und']['0']['filename'];
```

```
$wactual = $user_fields->field_fotografia['und']['0']['width'];
```

```
$hactual = $user_fields->field_fotografia['und']['0']['height'];
```

```
//obtenir nom associat a DNI
```

· S'obté el nom associat a aquest DNI o NIE.

```
$queryc = db_select('client');
```

```
$queryc->fields('client', array('nif', 'nom', 'cognom1', 'cognom2', 'email'))
->condition('nif', $nifactual, '=');
```

```
$resultc = $queryc->execute();
$recordc = $resultc->fetchAssoc();
```

//obtenir quantia logros associats a DNI

- S'obté el nombre de fites associades a aquest DNI.

```
$queryl = db_select('logrouser');
$queryl->fields('logrouser', array('nif', 'numlogro'))
->condition('nif', $nifactual, '=');
```

```
$resultl = $queryl->execute();
$count_resultl = $resultl->rowCount();
$recordl = $resultl->fetchAssoc();
```

```
$querylt = db_select('logro');
$querylt->fields('logro', array('nom'));
```

```
$resultlt = $querylt->execute();
$count_resultlt = $resultlt->rowCount();
```

- S'estableix el nivell de l'usuari en base a la *ratio* fites obtingudes / fites totals en el sistema. Això assegura variacions continuades en el nivell, i que per mantenir-lo s'hagi de participar.

```
//establir nivell del usuari
if (($count_resultl/$count_resultlt)<= '0.25'){
    $nivell='Principiant';
}
else if (($count_resultl/$count_resultlt) >= '0.75'){
    $nivell='Avançat';
}
else if (($count_resultl/$count_resultlt) == '1'){
```

```
$nivell='Hardcore';
}
else{
$nivell='Normal';
}
```

- Es conta els dies que ha assistit l'usuari i es guarda la data més nova.

```
// Contar dies assistits

$queryas = db_select('assistencia');

$queryas->fields('assistencia', array('nom', 'cognom1', 'cognom2', 'dass',
'mass', 'aass'))

->condition('nom', $recordc['nom'], '=')
->condition('cognom1', $recordc['cognom1'], '=')
->condition('cognom2', $recordc['cognom2'], '=')
->orderBy('aass', 'DESC')
->orderBy('mass', 'DESC')
->orderBy('dass', 'DESC');

$resultas = $queryas->execute();

$count_resultas = $resultas->rowCount();

$recordas = $resultas->fetchAssoc();
```

- S'inicialitza la variable que retornarà a la constructora. Es crea una taula per posar el nom, la imatge, el nivell, el nombre de fites obtingudes i l'últim dia d'assistència al centre.

```
$output = '<table>

<th><h0>'.$recordc['nom'].' '.$recordc['cognom1'].'
'.$recordc['cognom2'].'</h0></th>

<th><h0>Nivell: '.$nivell.'</h0><br>(Per a pujar de nivell,
completa més fites)</th>

<tr>

<td></td>

<td><p><h3>Fites assolides:
'.$count_result1.'/'.$count_result1t.'</h3></p>

<p><h3>Dies assistits: '.$count_resultas.'</h3></p>

<p><h3>Última Assistència:
'.$recordas['dass'].'/'.$recordas['mass'].'/'.$recordas['aass'].'</h3></p>
```

```
</td>
<br></table>';
```

```
$output .= '<br><br><h3>Fites Assolides: </h3>';
```

```
$output .=
'<table><th><h3>Medalla</h3></th><th><h3>Fita</h3></th><th><h3>Descripció</h3>
</th><tr>';
```

```
//MEDALLES
```

- Les primeres medalles són les variables, que es poden perdre en estar temps sense interactuar amb el sistema o que es donen automàticament en complir alguna feta.

```
$mes=date('n');
```

```
$any=date('Y');
```

- En cas de no haver assistit en el mes actual al centre, s'atorga la fita *Missing in action*.

```
$querynoas = db_select('asistencia');
$querynoas->fields('asistencia', array('nom', 'cognom1', 'cognom2',
'dass', 'mass', 'aass'))
->condition('nom', $recordc['nom'],'=')
->condition('cognom1', $recordc['cognom1'],'=')
->condition('cognom2', $recordc['cognom2'],'=')
->condition('mass', $mes,'=')
->condition('aass', $any,'=');

$resultnoas = $querynoas->execute();
$count_resultnoas = $resultnoas->rowCount();

if($count_resultnoas == 0){
    $output .= '<td></td>
<td>Missing In Action</td>
<td>No has aparegut pel gimnàs aquest mes!</td><tr>';
```

```
}
```

- En cas d'haver aparegut només un cop en el mes actual, s'atorga la fita *Punta de fletxa*.

```
if($count_resultnoas == 1){
    $output .= '<td></td>
    <td>Punta de fletxa</td>
    <td>Has vingut un dia per veure que tal esta la
cosa...</td><tr>';
}
```

- En cas de haver assistit en el mes actual més dies dels que queden per finalitzar-lo, s'atorga la fita *Duracell*.

```
if($count_resultnoas > 15){
    $output .= '<td></td>
    <td>Duracell</td>
    <td>Aquest mes has vingut més dies dels que queden per
finalitzar-lo!</td><tr>';
}
```

- Si el teu nivell es Normal, s'atorga la fita *Moving!* amb medalla normal.

```
if($nivell == 'Normal'){
    $output .= '<td></td>
    <td>Moving!</td>
    <td>Has completat més del 25% de les fites!
Enhorabona!</td><tr>';
}
```

- Si el teu nivell es Principiant, s'atorga la fita *Moving!* amb medalla bàsica.

```
if($nivell == 'Principiant'){
    $output .= '<td></td>
```

```

        <td>Moving!</td>
        <td>Has completat menys del 25% de les fites!
Enhorabona!</td><tr>';
    }

```

- Si el teu nivell es Avançat, s'atorga la fita *Moving!* amb medalla avançada.

```

    if($nivell == 'Avançat'){
        $output .= '<td></td>
        <td>Final!</td>
        <td>Has completat més del 75% de les fites!
Enhorabona!</td><tr>';
    }

```

- Si el teu nivell es Hardcore, s'atorga la fita *Final!*.

```

    if($nivell == 'Hardcore'){
        $output .= '<td></td>
        <td>Final!</td>
        <td>Has completat totes les fites! Enhorabona!</td><tr>';
    }

```

- Les segones medalles són les fixes. Les dona un usuari entrenador en assolir certes fetes i no es poden perdre.

```
//llistat de fites obtingudes.
```

- Es selecciona les fites de la taula d'associacions en les que està reflexat l'usuari actual.

```

$queryfit = db_select('logrouser');
$queryfit->fields('logrouser', array('nif', 'numlogro'))
->condition('nif', $nifactual, '=');

$resultfit = $queryfit->execute();

```

```
$count_resultfit = $resultfit->rowCount();
```

- Es mostren ordenades en una taula, amb la medalla i l'explicació de la fita que s'ha assolit amb ella.

```
while ($recordfit = $resultfit->fetchAssoc()){

    $querylog = db_select('logro');

    $querylog->fields('logro', array('num', 'nom',
'descripcio', 'nomimg'))

    $recordfit['numlogro'], '=')) ->condition('num',

    $resultlog = $querylog->execute();

    $recordlog = $resultlog->fetchAssoc();

    $output .= '<td></td>

    <td>'. $recordlog['nom'] . '</td>

    <td>'. $recordlog['descripcio'] . '</td><tr>';

}

$output .= '</table>';
```

- Per a finalitzar, els usuaris poden veure quines fites encara no han assolit.

```
$output .= '<br><br><h3>Fites Pendants: </h3>';

$output .=
'<table><th><h3>Medalla</h3></th><th><h3>Fita</h3></th><th><h3>Descripció</h3>
</th><tr>';
```

- Es selecciona les fites de la taula d'associacions en les que no està reflexat l'usuari actual.

```
//llistat de fites no obtingudes.
```

```
$querynfit = db_select('logrouser');

$querynfit->fields('logrouser', array('nif', 'numlogro'))

->condition('nif', $nifactual, '=');
```

```
$resultnfit = $querynfit->execute();
```

- Es mostren ordenades en una taula, amb la medalla i l'explicació de la fita que s'ha assolit amb ella.

```
while ($recordnfit = $resultnfit->fetchAssoc()){

    $querynlog = db_select('logro');
    $querynlog->fields('logro', array('num', 'nom',
'descripcio', 'nomimg'))
    ->condition('num',
$resultnfit['numlogro'],'<>');

    $resultnlog = $querynlog->execute();
    $recordnlog = $resultnlog->fetchAssoc();

    $output .= '<td></td>
<td>'.$recordnlog['nom'].'</td>
<td>'.$recordnlog['descripcio'].'</td><tr>';

}

$output .= '</table>';
return $output;
}
```

4.3 Disseny de pàgines estàtiques que compleixen funcionalitats:

Per a realitzar la tasca de mostrar les dades de clients que han assistit en el dia actual, es crea un tipus de contingut *assisttoday* en el que es mostraran les dades d'assistència. Just davall de la constructora de *content*, es situa el codi *php* següent:

```
<?php
```

- Agafa els valors del dia actual

```
$dia=date('j');
```

```
$mes=date('n');
```



```
$any=date('Y');
```

- Realitza una cerca en la taula assistència, recaptant tots els valors associats al dia d'avui i ordenant-los de més aviat a més tard.

```
$query = db_select('assistencia');  
  
$query->fields('assistencia', array('nom', 'cognom1', 'hass', 'mmass',  
'dass', 'mass', 'aass'))  
  
->condition('dass', $dia, '=')  
->condition('mass', $mes, '=')  
->condition('aass', $any, '=')  
->orderBy('hass', 'ASC') //ORDER BY contador  
->orderBy('mmass', 'ASC');  
  
$result = $query->execute();
```

- Inicialitza una taula on mostrar els resultats.

```
echo '<table>  
<th>Nom</th>  
<th>Cognom</th>  
<th>Hora</th></tr>';
```

- Inclou en la taula tots els resultats d'assistència del dia d'avui.

```
while ($record = $result->fetchAssoc()){  
    echo '<tr>  
    <td>'.$record['nom'].'</td>  
    <td>'.$record['cognom1'].'</td>  
    <td>'.$record['hass'].' Hores '.$record['mmass'].' Minuts</td></tr>';  
  
    }  
  
    echo '</table>'; ?>  
<?php endif; ?>
```

Bloc 5:

Conclusions

La realització d'aquest projecte (tenint en compte que els alumnes de sistemes no reben formació en temes d'usabilitat) ha resultat un bon complement a l'aprenentatge rebut durant la carrera.

La primera quinzena de setembre serà un període crític per a aquesta aplicació, en que degut a la decisió de re-organitzar totes les fitxes de clients (per part del centre) probablement seran necessaris canvis en el disseny i estructura de l'apartat administratiu.

Lluny d'haver finalitzat, aquest projecte entra en la fase dos, en la que es desenvoluparà la temàtica de gamificació, englobant grups d'usuaris i afegint categories. També hi ha intenció d'estendre aquesta pràctica a altres centres, entre els quals es podrà posar en comú la informació de les diverses competicions, on els clients seran els representants dels diferents gimnasos, com en els antics *ludus* romans (però per voluntat pròpia). El mateix centre servirà de publicitat en les seves reunions amb altres gimnasos i pel boca a boca.

Per últim, s'està esperant a la apertura del centre després del període de vacances, per tal de poder comprovar els resultats d'aquest projecte en un segment de temps relativament llarg i amb una càrrega d'usuaris més elevada.

5.1 Usabilitat:

Havent acabat el treball, es duen a terme un seguit de testos que per comprovar com interacciona el resultat amb diferents sistemes operatius, navegadors, i persones.

5.1.1 Navegadors:

S'utilitza el servei web de testatge de navegadors <http://browsershots.org/>, des d'on retorna una captura de pantalla dels

navegadors (màquines amb S.O. *Linux*): *Opera* 12.02, *chrome* 28, *firefox* 23, *Epiphany* 3.6.1, *konqueror* 4.9. (amb un resultat visual correcte, tot i que hi ha petits canvis en els colors resultants, figures 42 i 43).

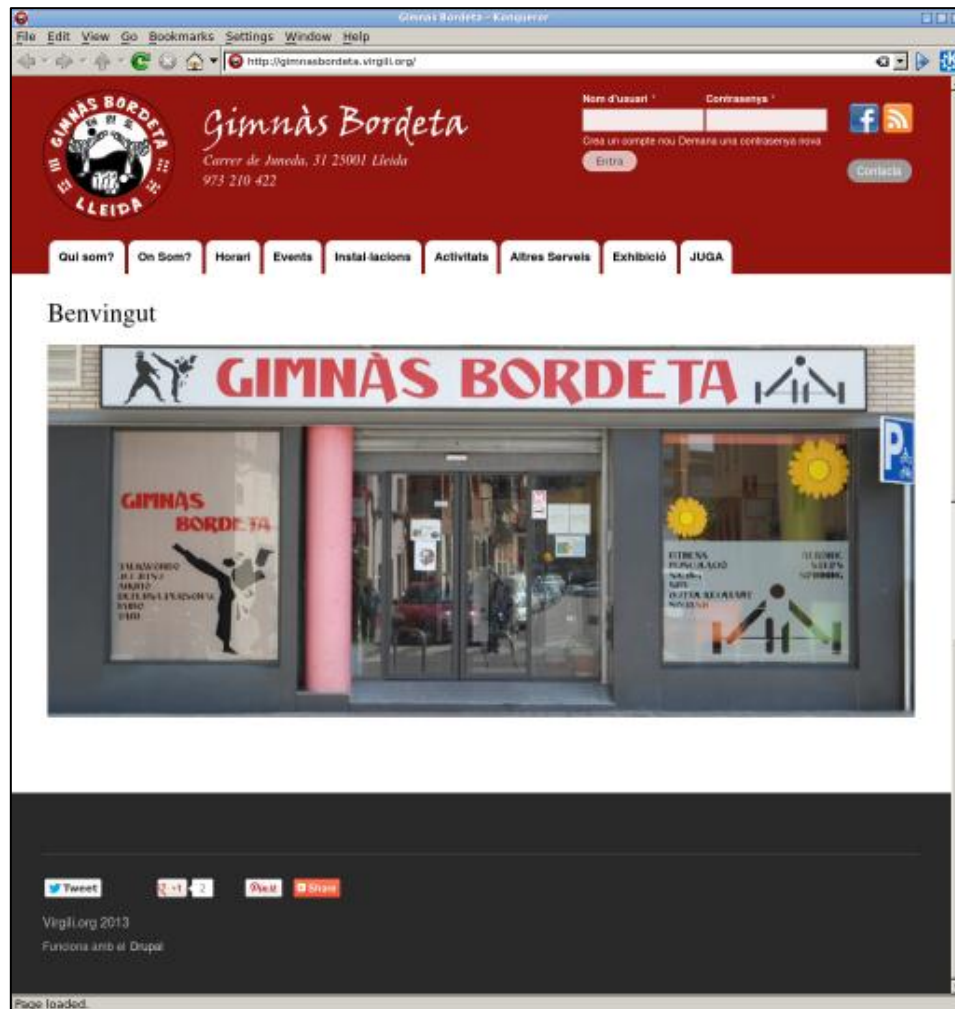


Figura 42: Navegadors Web (Konqueror)

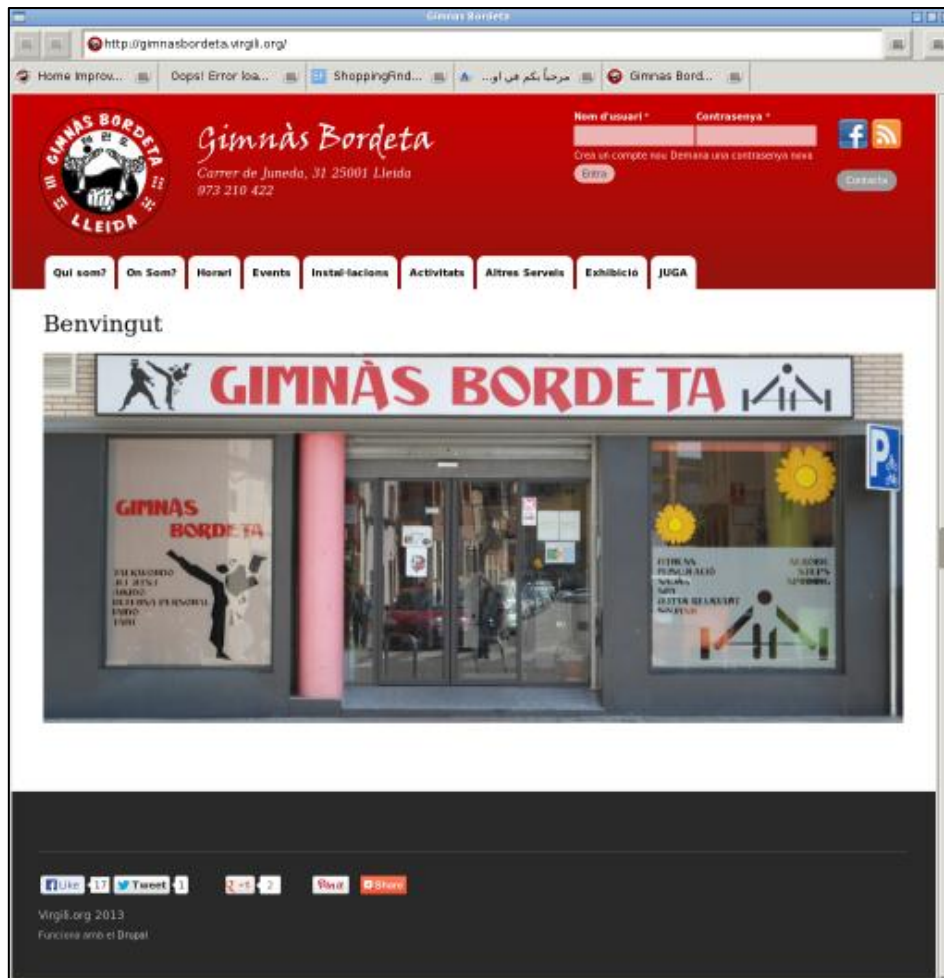


Figura 43: Navegadors Web (Epiphany)

En Windows tant XP, com 7 i 8, en *chrome* i *firefox* 23 el seu funcionament és exactament el desitjat (ja que durant la fase de implementació, s'ha anat comprovant la seva funcionalitat en aquestes plataformes).

Per comprovar la interactivitat mitjançant *Internet Explorer* s'utilitza el servei web <http://netrender.com> que permet seleccionar la versió de IE en la que retorna la captura de pantalla. De millor a pitjor, IE 10 no és capaç de mostrar la tipologia del logotip, però totes les funcionalitats són correctes; IE 9 té la fallada de IE 10, però afegim que no es capaç d'utilitzar els degradats de colors correctament. Per últim, la versió més antiga de IE, la versió 7 (que es troba inclosa en el paquet d'aplicacions de *Windows XP*), té les mateixes fallides que les versions 9 i 10, però no agafa correctament els logotips, els *css* estilitzats són ignorats (generant botons quadrats en comptes de rodons) i en general les imatges no es carreguen bé, ja que són solapades per la descripció (qualitat bàsica avui en dia). No obstant, en el pitjor dels casos continua sent funcional, i per tant es troba un

resultat satisfactori. Per últim es prova en el *Internet Explorer* per a *Xbox360* (que a mode d'anècdota, no és capaç de carregar la web de *Facebook* ni *Youtube*) i funciona igual de bé que en un PC amb *Firefox 23*.



Figura 44: Internet Explorer 10



Figura 45: Internet Explorer 9

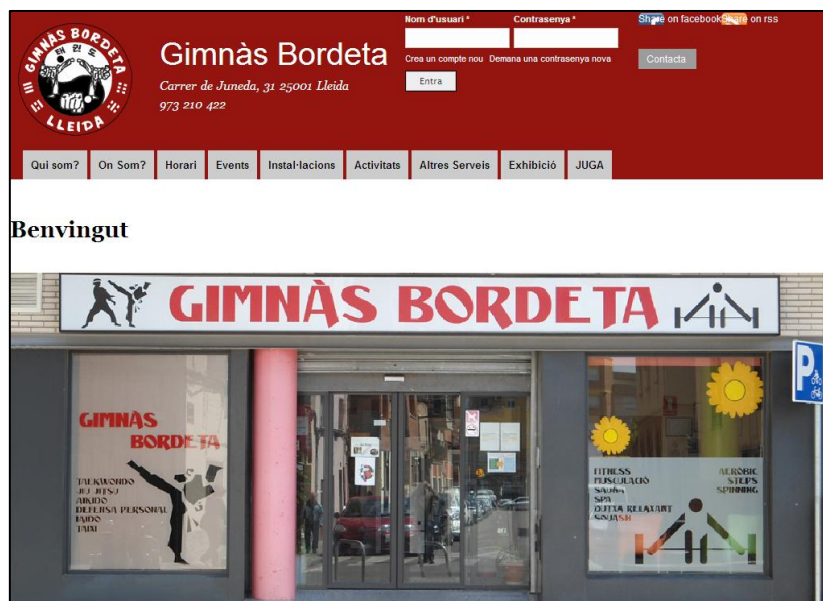


Figura 46 Internet Explorer 7

5.1.2 TAW:

Sigles de Test d'Accessibilitat Web, que serveix per comprovar l'execució potencial de l'aplicatiu web independentment del sistema operatiu o navegador.

En concret s'analitza el compliment de les *Web Content Accessibility Guidelines* (WCAG), que componen un conjunt de directrius per fer el contingut accessible, sobretot per als usuaris amb discapacitat, incloent també totes les aplicacions d'usuari i dispositius altament limitats (com els telèfons mòbils). La versió actual, 2.0, és també una norma ISO. L'aplicació d'usabilitat utilitzada per aquest fi ha estat l'eina web <http://www.tawdis.net>, que realitza un test WCAG2.0. El resultat ha estat un problema robust i 66 advertències:

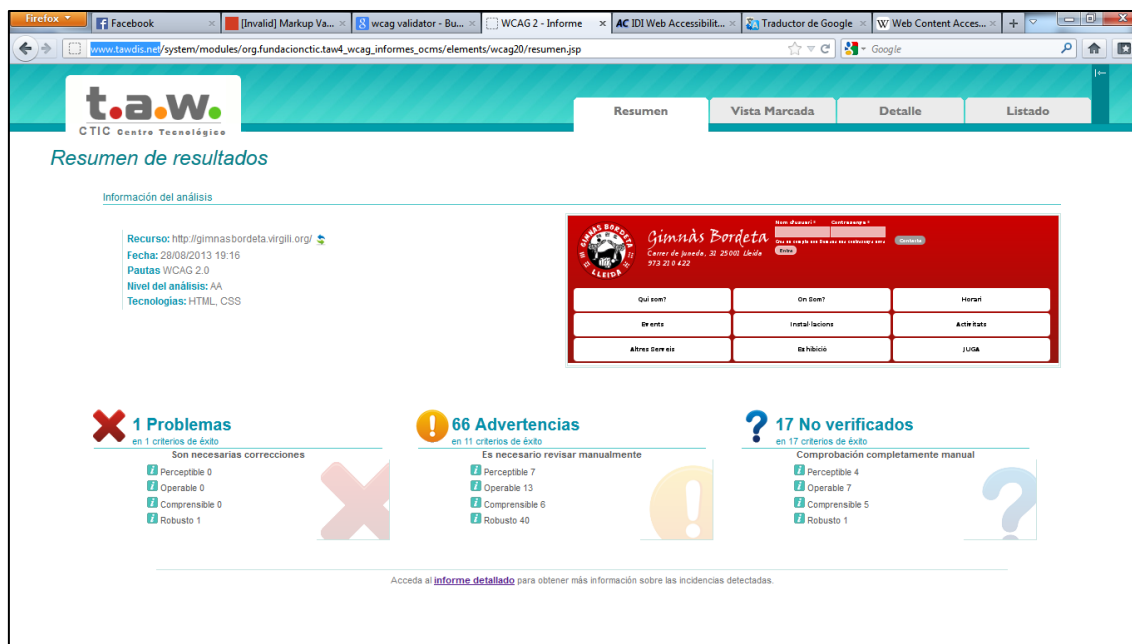


Figura 47: Anàlisi WCAG2.0

El problema robust es solventa ràpid, ja que únicament es un `
` que (com es obvi) no s'ha tancat. Tot i que no és necessari tancar les etiquetes úniques, les normes de bona praxis del format fan necessari tancar l'element en la mateixa declaració `
`.

La resta d'advertències resulten ser d'assistència en dispositius per a discapacitats, com per exemple les descripcions de les imatges, que la pàgina funcioni igual sense necessitat de *css* o l'ús de *Javascripts*.

5.1.2 Usuaris:

Per últim, es realitzen un seguit de proves d'utilització del portal amb usuaris, que són gravades en video. Les eines d'administració també s'han testat amb aquests usuaris. Els resultats es troben adjunts en el disc.

Bloc 6:

Bibliografia

Aquesta és la bibliografia que ha ajudat en el desenvolupament d'aquest projecte:

Llibres:

- Per a organitzar les fases del projecte i realitzar un correcte prototipat:

“Diseño de sistemas interactivos centrados en el usuario” (Granollers, Toni; Lorés, Jesús; Cañas, José; 2005)

- Per a aplicar un pensament de joc en la vida real i dissenyar-lo després:

“For the Win (how game thinking can revolutionize your business)” (Werbach, Kevin; Hunter, Dan; 2012)

“Gamification by Design (Implementing Game Mechanics in Web and Mobile Apps)” (Zichermann, Gabe; Cunningham, Christopher; 2011)

Revistes:

- Per estar a la última en temes de disseny:

“Web Designer Magazine (UK Edition)” (www.webdesignermag.com.uk, números de 2012 a 2013 escrits)

Webs:

- Tot el necessari per entendre com interactuar amb *Drupal* i els seus mòduls:

“Official Drupal API” (<https://api.drupal.org>)

- Per desenvolupar el portal:

“CS Designer” (tanicek, Petr; <http://colorshemesdesigner.com>, 2010)

“Contrast A” (Hamann, Annika; <http://www.dasplankton.de/ContrastA/>, 2009)

Annex A: Enquestes realitzades

Qüestionari per als usuaris



Edat de qui realitza el qüestionari:

- ☐ menys de 15 anys
- ☐ entre 15 i 18 anys
- ☐ entre 18 i 25 anys
- ☐ entre 25 i 45 anys
- ☐ més de 45 anys

En quina franja horària assisteix al gimnàs:

- ☐ Matí
- ☐ Tarda

En quines de les activitats de les que ofereix el centre participes (es pot marcar més d'una opció):

- | | |
|-----------------------------------|---|
| <input type="checkbox"/> Gimnàs | <input type="checkbox"/> Ioga |
| <input type="checkbox"/> Aeròbic | <input type="checkbox"/> Defensa Personal |
| <input type="checkbox"/> Spinning | <input type="checkbox"/> Aikido |
| <input type="checkbox"/> Tai Chi | <input type="checkbox"/> Taekwondo |
| <input type="checkbox"/> Pilates | <input type="checkbox"/> Squash |

Quant de temps fa que vens al centre (en anys)?

- | | | |
|---|---------------------------------|--|
| <input type="checkbox"/> Menys de 1 Any | <input type="checkbox"/> 1 Any | <input type="checkbox"/> 2 Anys |
| <input type="checkbox"/> 3 Anys | <input type="checkbox"/> 4 Anys | <input type="checkbox"/> Més de 5 Anys |

Quines de les següents motivacions per venir al centre s'ajusten a les teves:

- | | |
|--------------------------------------|--|
| <input type="checkbox"/> Recuperació | <input type="checkbox"/> Augmentar la musculació |
|--------------------------------------|--|

- | | |
|---|--|
| <input type="checkbox"/> Mantenir la forma | <input type="checkbox"/> Preparar competicions |
| <input type="checkbox"/> Preparació per a proves d'accés | <input type="checkbox"/> _____ |
| <input type="checkbox"/> Preparació per a un tipus de treball | <input type="checkbox"/> _____ |
| <input type="checkbox"/> Realitzar alguna de les Activitats | <input type="checkbox"/> _____ |

Has entrat algun cop a la web del gimnàs?

- ☐ Sí ☐ No

Que hi busques?

- | | |
|--|--|
| <input type="checkbox"/> Horaris | <input type="checkbox"/> Informació sobre Competicions |
| <input type="checkbox"/> Informació sobre Activitats | <input type="checkbox"/> _____ |
| <input type="checkbox"/> Informació sobre preus | <input type="checkbox"/> _____ |

T'agradaria poder reservar els equipaments per internet (com la sala d'esquaix o la sauna)?

- ☐ Sí ☐ No

Navegues habitualment per Internet amb el telèfon mòbil? En cas afirmatiu, quin telèfon tens:

- | | |
|-------------------------------------|---|
| <input type="checkbox"/> HTC | <input type="checkbox"/> Samsung Galaxy |
| <input type="checkbox"/> Blackberry | <input type="checkbox"/> Nokia |
| <input type="checkbox"/> iPhone | <input type="checkbox"/> _____ |

Quines de les següents xarxes socials utilitzes:

- | | |
|------------------------------------|----------------------------------|
| <input type="checkbox"/> Facebook | <input type="checkbox"/> Twitter |
| <input type="checkbox"/> Instagram | <input type="checkbox"/> Tuenti |
| <input type="checkbox"/> _____ | |

Has entrat algun cop a la pàgina de Facebook del gimnàs?

☐ Sí ☐ No

T'agradaria trobar-hi:

- | | |
|--|--|
| <input type="checkbox"/> Horaris | <input type="checkbox"/> Fotografies |
| <input type="checkbox"/> Informació sobre activitats | <input type="checkbox"/> Afegir-te a la comunitat del Gimnàs |
| <input type="checkbox"/> Informació sobre preus | <input type="checkbox"/> _____ |

Observacions:

Cuestionario para los usuarios

Edad de quien realiza el cuestionario:

- ☐ menos de 15 años
- ☐ entre 15 y 18 años
- ☐ entre 18 y 25 años
- ☐ entre 25 y 45 años
- ☐ mas de 45 años



En quina franja horaria asistes al gimnasio:

- ☐ Mañanas
- ☐ Tardes

En cuáles de las actividades de las que ofrece el centro participas (se puede marcar más de una opción):

- | | |
|-----------------------------------|---|
| <input type="checkbox"/> Gimnasio | <input type="checkbox"/> Yoga |
| <input type="checkbox"/> Aeróbic | <input type="checkbox"/> Defensa Personal |
| <input type="checkbox"/> Spinning | <input type="checkbox"/> Aikido |
| <input type="checkbox"/> Tai Chi | <input type="checkbox"/> Taekwondo |
| <input type="checkbox"/> Pilates | <input type="checkbox"/> Squash |

¿Cuánto tiempo hace que vienes al centro (en años)?

- | | | |
|---|---------------------------------|--|
| <input type="checkbox"/> Menos de 1 Año | <input type="checkbox"/> 1 Año | <input type="checkbox"/> 2 Años |
| <input type="checkbox"/> 3 Años | <input type="checkbox"/> 4 Años | <input type="checkbox"/> Más de 5 Años |

Cuáles de las siguientes motivaciones para venir al centro se ajustan a las tuyas:

- | | |
|--|--|
| <input type="checkbox"/> Recuperación | <input type="checkbox"/> Aumentar la musculación |
| <input type="checkbox"/> Mantener la forma | <input type="checkbox"/> Preparar competiciones |
| <input type="checkbox"/> Preparación para pruebas de acceso | <input type="checkbox"/> _____ |
| <input type="checkbox"/> Preparación para un tipo de trabajo | <input type="checkbox"/> _____ |

☐ Realizar alguna de las Actividades ☐ _____

Has entrado alguna vez en la web del gimnasio?

☐ Si ☐ No

¿Qué buscabas?

☐ Horarios ☐ Información sobre competiciones
☐ Información sobre Actividades ☐ _____
☐ Información sobre precios ☐ _____

¿Te gustaría poder reservar los equipamientos por internet (como la sala de squash o la sauna)?

☐ Si ☐ No

¿Navegas habitualmente por Internet con el teléfono móvil? En caso afirmativo, qué teléfono usas:

☐ HTC ☐ Samsung Galaxy
☐ Blackberry ☐ Nokia
☐ iPhone ☐ _____

Cuáles de las siguientes redes sociales utilizas:

☐ Facebook ☐ Twitter
☐ Instagram ☐ Tuenti
☐ _____

¿Has entrado alguna vez en la página de Facebook del gimnasio?

☐ Si ☐ No

Te gustaría encontrar:

☐ Horarios ☐ Fotografías

- | | |
|--|---|
| <input type="checkbox"/> Información sobre actividades | <input type="checkbox"/> Sumarte a la comunidad |
| <input type="checkbox"/> Información sobre precios | <input type="checkbox"/> _____ |

Observaciones:
